



Universidad  
Carlos III de Madrid

Departamento de Informática

PROYECTO FIN DE GRADO

Aplicación móvil para fomentar  
la seguridad vial en niños  
(Plataforma IOS)

Autor: Álvaro García Reyes

Tutor: Javier García Guzmán

Cotutor: Alberto Heredia García

Leganés, Septiembre de 2012



Título: Aplicación móvil para fomentar la seguridad vial en niños (plataforma IOS)  
Autor: Álvaro García Reyes  
Director: Javier García Guzmán

EL TRIBUNAL

Presidente: \_\_\_\_\_

Vocal: \_\_\_\_\_

Secretario: \_\_\_\_\_

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día \_\_\_\_ de \_\_\_\_\_ de 20\_\_ en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE



# Agradecimientos

En primer lugar quería agradecer el apoyo de mi familia durante toda la carrera ya que sin ellos no podría haber finalizado esta etapa de la vida. Agradecer a mi padre el apoyo en mis estudios. A mi madre, por ser quién es, por apoyarme en los momentos más difíciles, por entender mis inquietudes y por ser Madre y amiga. Y por último a mi hermano que ha sabido comprenderme en todo momento y ha sacrificado mucho para que yo pueda estar donde estoy.

También destacar a los compañeros de carrera que durante los años en la universidad he compartido todo con ellos y que con su apoyo y su ayuda he podido llegar hasta aquí. Gracias a Ignacio García Mediavilla, Fernando García Albaladejo, Santiago García Delgado, Marta Garrido Iglesias, Sergio Godoy Muñoz -Torrero, Gonzalo Herranz Martínez de Pinillos, Yolanda León Román, Borja López Gómez y especial mención para Álvaro García Martínez, el cual me ha apoyado incondicionalmente durante estos cuatro años.

Agradezco a mi tutor, Javier García Guzmán, el apoyo constante, la confianza que ha depositado en mí y los múltiples consejos que me ha ofrecido para que este proyecto se pueda llevar a cabo. También destacar a los profesores Alberto Heredia García y Juan Carlos Alonso por los documentos proporcionados y las propuestas realizadas.

Por otro lado agradecer a Adrián Casado Rivas, Javier Sánchez Hernández, Iván Sánchez Perea, Soledad Díaz Carrizo y Roberto Esteban Santiago, mis compañeros de despacho, el apoyo y la resolución de dudas durante todo el proyecto.

Cabe destacar la comprensión y la paciencia de Alicia Fernández Gallardo en este último año tanto en temas personales como universitarios. Además agradezco a todas las personas ajenas a la universidad el apoyo ofrecido para que pudiera crecer como persona y como estudiante.

Para finalizar quería dar las gracias a, mis dos alumnos, Tomás Catalán Caballero (nueve años) y a Carlos Catalán Caballero (seis años) por ayudarme a probar la aplicación en sus últimas etapas.



# Resumen

La enseñanza de la seguridad vial hoy en día es fundamental para concienciar y evitar posibles situaciones peligrosas. El cumplimiento de las normas de seguridad vial disminuiría este tipo de situaciones.

El problema que resuelve este proyecto es la dificultad de enseñar ciertas normas básicas de seguridad vial a los niños. En este caso concreto se ha enfocado a los niños más pequeños, de una edad entre seis y diez años.

Este problema junto con la importancia que está alcanzando el uso de las tecnologías móviles y el uso cada vez más temprano de los niños a estas tecnologías, es lo que motiva la realización de este TFG.

El objetivo principal de este proyecto es fomentar la seguridad vial en los niños. Para ello se ha creído conveniente crear un juego para captar la atención del usuario y poder entretener a la vez que aprender las normas de seguridad vial básicas.

La metodología de trabajo que se ha utilizado durante la realización del proyecto ha sido una metodología ágil de desarrollo: Scrum. Se ha elegido esta opción porque se adaptaba bien a la planificación adaptativa que se ha seguido a lo largo de todo el proyecto.

Se ha elegido la plataforma IOS como plataforma de desarrollo, y se han utilizado herramientas para el desarrollo, como son el lenguaje de programación Objective-C y la herramienta Xcode.

Como resultado, se ha obtenido una aplicación móvil que ofrece a los usuarios, los niños, un juego interactivo por medio de láminas de errores, en la cual estos usuarios intentarán acertar las posibles incidencias que haya en ellas y así aprenderán las normas de seguridad vial. Cabe destacar que a medida que se ha ido implementando la aplicación se ha ido probando con niños de seis y nueve años.

**Palabras clave:** Seguridad vial, niños, iPhone, IOS.





# Abstract

Teaching road safety is really important today to raise awareness and avoid dangerous situations. Fulfilling with the road safety rules would decrease those situations.

The problem that solves is the difficulty of teaching certain basic road safety rules to children. In this particular case it has been focused on younger children, between six and ten years old.

Besides the importance reaching of the use of mobile technologies and the earlier use of these technology by the children, is what motivates the realization of this TFG.

The main target of this project is to promote road safety in children. For this reason, this game has been created, with the intention of attract the user's attention and to entertain while learning the basic road safety rules.

The methodology that has been used for the project is an agile development methodology: Scrum. We have chosen this option because it was well suited to adaptive planning that has been followed throughout the project.

The IOS platform was chosen as development platform. We have used development tools, such as the programming language Objective-C and Xcode tool.

As a result, we have obtained a mobile application that provides to the users, the children, an interactive game using pictures of errors, where these users will try to ascertain the possible impact they have of them, so that, they will learn the rules of road safety. It is remarkable that meanwhile the application has been implementing, it has been tested with six and nine years old children.

**Keywords:** road safety, children, iPhone, IOS.

# Índice general

<b>1. CAPÍTULO 1 INTRODUCCIÓN .....</b>	<b>15</b>
1.1 Motivación del proyecto .....	16
1.2 Objetivos .....	16
1.3 Método de resolución .....	17
1.4 Medios empleados.....	18
1.5 Terminología .....	18
1.6 Estructura de la memoria .....	22
<b>2. CAPÍTULO 2 ESTADO DEL ARTE.....</b>	<b>23</b>
2.1 Criterios de búsqueda .....	24
2.2 Análisis de las aplicaciones relacionadas con los objetivos establecidos .....	25
2.3 Entorno tecnológico seleccionado.....	33
2.4 Método para realizar el proyecto: Desarrollo Ágil .....	41
<b>3. CAPÍTULO 3 ANÁLISIS DE REQUISITOS.....</b>	<b>45</b>
3.1 Descripción general.....	46
3.2 Requisitos de usuario .....	46
3.3 Casos de uso.....	48
3.4 Requisitos de software .....	50
<b>4. CAPÍTULO 4 DISEÑO .....</b>	<b>55</b>
4.1 Diseño externo .....	56
4.2 Diseño interno .....	63
<b>5. CAPÍTULO 5 PLAN DE PRUEBAS .....</b>	<b>77</b>
5.1 Pruebas de aceptación .....	78
<b>6. CAPÍTULO 6 GESTIÓN DEL PROYECTO .....</b>	<b>84</b>
6.1 Planificación del proyecto.....	85
6.2 Análisis económico .....	89
<b>7. CAPÍTULO 7 CONCLUSIONES .....</b>	<b>93</b>
7.1 Conclusiones del proyecto .....	94
7.2 Conclusiones personales .....	96
<b>8. BIBLIOGRAFÍA.....</b>	<b>98</b>

# Índice de figuras

Ilustración 1 - Menús de confort auto .....	26
Ilustración 2 - Mapfre juego .....	27
Ilustración 3 - RACE menús .....	28
Ilustración 4 - RACE videos .....	28
Ilustración 5- Educapeques .....	29
Ilustración 6 - Scoda.....	30
Ilustración 7 - DGT juego de foro .....	31
Ilustración 8 - DGT lámina de errores .....	31
Ilustración 9 - DGT mapa .....	32
Ilustración 10- Comparativa del mercado de Smartphone por compañías.....	34
Ilustración 11 - Comparativa de ventas.....	35
Ilustración 12 - Logotipo de IOS .....	36
Ilustración 13 - Icono de Xcode .....	37
Ilustración 14 - Icono de Adobe photoshop .....	38
Ilustración 15 - Icono de Microsoft Office .....	39
Ilustración 16 - Icono de StarUML .....	39
Ilustración 17 - Icono de GanttProject .....	40
Ilustración 18 - Icono de Omnigraffle profesional.....	40
Ilustración 19 - Diagrama de flujo de Scrum .....	43
Ilustración 20 - Casos de uso .....	49
Ilustración 21 - Diseño inicial menú .....	57
Ilustración 22 - Diseño inicial imagen .....	57
Ilustración 23 - Diseño inicial pregunta .....	58
Ilustración 24 - Diseño inicial error .....	59
Ilustración 25 - Diseño interno felicitaciones .....	59
Ilustración 26 - Diseño final menú .....	61
Ilustración 27 - Diseño final tutorial .....	61
Ilustración 28 - Diseño final lámina.....	62
Ilustración 29 - Diseño final felicidades .....	62
Ilustración 30 - Diagrama de Secuencia CU-02.....	63
Ilustración 31 - Diagrama de Secuencia CU-03.....	64
Ilustración 32 - Diagrama de secuencia CU-01 .....	64
Ilustración 33 - Diagrama de clases .....	66
Ilustración 34 - Proceso de creación .....	68
Ilustración 35 - Icono de la aplicación .....	68
Ilustración 36 - Personajes plantilla1 .....	69
Ilustración 37 - Personajes plantilla2 .....	70
Ilustración 38 - Personajes plantilla3 .....	70
Ilustración 39 – Fondo plantilla1 .....	71
Ilustración 40 - Fondo plantilla2 .....	71
Ilustración 41 - Fondo plantilla3 .....	72
Ilustración 42 - Plantilla1 .....	72
Ilustración 43 - Plantilla2 .....	73
Ilustración 44 - Plantilla3 .....	73
Ilustración 45 - Menú 3 nivel alto .....	74
Ilustración 46 - Felicitaciones 1 .....	75

Ilustración 47 - Felicitaciones 2 ..... 75

Ilustración 48 - Felicitaciones 3 ..... 75

Ilustración 49 - Diagrama de Gantt inicial ..... 86

Ilustración 50 - Diagrama de Gantt final..... 88

# Índice de tablas

Tabla 1- Comparación de aplicaciones .....	33
Tabla 2- Comparación entre Metodología Ágil y Tradicional .....	42
Tabla 3 - Plantilla de ejemplo de requisitos de usuario.....	46
Tabla 4 - RUC-01 Seleccionar lámina .....	47
Tabla 5 - RUC-02 Ver tutorial .....	47
Tabla 6 - RUC-03 Interactuar con la lámina .....	47
Tabla 7 - RUC-04 Cambiar lámina .....	47
Tabla 8 - RUC-05 Volver al menú principal.....	47
Tabla 9 - RUC-06 Ver puntuación .....	47
Tabla 10 - RUR-01 Idioma .....	48
Tabla 11 - RUR-02 Sistema en vertical .....	48
Tabla 12 - RUR-03 Minimizar tiempo de espera.....	48
Tabla 13 - RUR-04 Funcionamiento.....	48
Tabla 14- Plantilla de los casos de uso.....	49
Tabla 15- CU-01 Ver puntuación.....	49
Tabla 16- CU-02 Ver tutorial .....	50
Tabla 17- CU-03 Jugar.....	50
Tabla 18 - RSF-01 Mostrar imagen inicial .....	51
Tabla 19 - RSF-02 Extensibilidad del funcionamiento.....	51
Tabla 20 - RSF-03 Cambio de lámina.....	51
Tabla 21 - RSF-04 Cambio de lámina.....	51
Tabla 22 - RSF-04 Seleccionar lámina .....	51
Tabla 23 - RSF-06 Mostrar la opción tutorial .....	51
Tabla 24 - RSF-07 Ver tutorial .....	52
Tabla 25 - RSF-08 Cancelar tutorial .....	52
Tabla 26 - RSF-09 Mostrar lámina .....	52
Tabla 27 - RSF-10 Seleccionar incidencias .....	52
Tabla 28 - RSF-11 Mostrar un mensaje de incidencia.....	52
Tabla 29 - RSF-12 Mostrar el número de incidencias .....	52
Tabla 30 - RSF-13 Mostrar imagen de felicitación.....	52
Tabla 31 - RSF-14 Volver al menú principal.....	53
Tabla 32 - RSF-15 Terminar tutorial .....	53
Tabla 33 - RSF-16 Ver puntuación .....	53
Tabla 34 - RSNF-01 Idioma.....	53
Tabla 35 - RSNF-02 Orientación. ....	53
Tabla 36 - RSNF-03 Minimizar tiempo de espera .....	54
Tabla 37 - RSNF-04 Funcionamiento .....	54
Tabla 38 - Plantilla de Prueba de aceptación .....	78
Tabla 39 – PA-01 Abrir la aplicación .....	78
Tabla 40 – PA-02 Desplazarse en el menú .....	78
Tabla 41 – PA-03 Cambiar de lámina.....	79
Tabla 42 – PA-04 Seleccionar lámina.....	79
Tabla 43 – PA-05 Ver tutorial.....	79
Tabla 44 – PA-06 Cancelar tutorial .....	79
Tabla 45 – PA-07 Volver al menú principal Tutorial .....	79
Tabla 46 – PA-08 Seleccionar incidencia .....	80

Tabla 47 – PA-09 Aceptar mensaje .....	80
Tabla 48 – PA-10 Felicitaciones .....	80
Tabla 49 – PA-11 Volver al menú principal Felicitaciones .....	80
Tabla 50 – PA-05 Terminar tutorial .....	81
Tabla 51 - PA-13 Ver puntuación .....	81
Tabla 52 - Matriz RSF x PA .....	82
Tabla 53 - Resultado de las pruebas.....	83
Tabla 54 - Costes de personal .....	89
Tabla 55 - Coste de hardware.....	90
Tabla 56 - Coste de software.....	91
Tabla 57 – Costes indirectos .....	92
Tabla 58 – Coste Total .....	92

# Capítulo 1 Introducción

1.1 Motivación del proyecto .....	16
1.2 Objetivos .....	16
1.3 Método de resolución.....	17
1.4 Medios empleados.....	18
1.5 Terminología.....	18
1.5.1 Acrónimos y abreviaturas .....	18
1.5.2 Definiciones .....	19
1.6 Estructura de la memoria .....	22

## 1.1 Motivación del proyecto

La enseñanza de la seguridad vial hoy en día es fundamental para concienciar y evitar posibles situaciones peligrosas. El cumplimiento de las normas de seguridad vial disminuirá este tipo de situaciones. Además una buena educación es fundamental y más cuando se tratan de los más pequeños que son los que mayor capacidad de aprendizaje tienen y a su vez son los más difíciles de enseñar.

El problema que resuelve este proyecto es la dificultad de enseñar ciertas normas básicas de seguridad vial a los niños. En este caso concreto se ha enfocado a los niños más pequeños, de una edad entre seis y diez años.

Además en los tiempos que corren la tecnología con mayor potencial es la telefonía móvil. Cada año más gente da el salto al teléfono inteligente o Smartphone, ya que se ha quedado obsoleto que el teléfono móvil solo sirva para enviar mensajes y realizar llamadas. Las ventajas de disponer de uno de estos dispositivos son claras. Probablemente la mayor ventaja sean las innumerables y diversas aplicaciones que se pueden tener en estos dispositivos. Gracias a estas aplicaciones los usuarios pueden tener a su alcance diversas funciones como: realizar la compra, obtener entretenimiento, informarse de la noticias, entre muchas otras.

La importancia de enseñar a los niños seguridad vial para concienciarlos y evitar posibles situaciones de peligro junto con la importancia que está alcanzando el uso de las tecnologías móviles y el uso cada vez más temprano de los niños a estas tecnologías, es lo que motiva para realizar este proyecto, es decir, una aplicación móvil que fomente la seguridad vial en los niños.

## 1.2 Objetivos

El objetivo fundamental de este proyecto es el de diseñar e implementar una aplicación móvil para fomentar la seguridad vial a niños en la plataforma IOS. En base a dicho objetivo principal, se proponen los siguientes objetivos secundarios:

- Debe ser un juego.

Esta aplicación está orientada a los más pequeños y la mejor forma de captar la atención y comprensión de un niño es por medio de un juego divertido. En este caso gracias a las **láminas de errores** se podrá captar la atención de los usuarios y de una forma sencilla, estos podrán averiguar cuáles son las incidencias pudiendo así aprender las normas básicas de seguridad vial. De este objetivo se derivan otros dos más:



- La aplicación debe atraer la atención de los usuarios, los niños.

Este objetivo secundario es esencial ya que lo más difícil en los niños es captar su atención para que gracias a la aplicación pueda aprender las normas de seguridad vial.

- Debe ser sencillo para una buena comprensión.

De nada sirve captar la atención del usuario, si éste no comprende bien cómo utilizar la aplicación o las normas de seguridad vial que se le va a enseñar.

- La aplicación contenga diferentes niveles de dificultad

Este objetivo se ha creído necesario ya que fomenta la superación por parte del usuario. Además así se intentará llegar a un abanico más amplio de usuarios. Este abanico siempre está orientado a los niños. No será lo mismo que juegue un niño de seis años a que juegue uno de diez, por eso es imprescindible la diferencia de niveles.

- A ser posible que sea original.

Este objetivo es menos importante que los anteriores pero no se debe caer en lo cotidiano u ordinario, sino que debe sorprenderse desde un primer momento al usuario, que en este caso serán niños.

## 1.3 Método de resolución

El método llevado a cabo para la realización de la aplicación ha sido iterativo, ya que en un principio no se tenían claros los requisitos que debía cumplir la aplicación y se pretendía definirlos a lo largo del desarrollo del proyecto, dependiendo de los avances y opciones que fueran surgiendo.

Las iteraciones que se plantearon desde un inicio de este proyecto están basadas en las que se proponen en algunas de las metodologías ágiles, en las que las reuniones toman una importancia considerable. La frecuencia de las reuniones se estableció al inicio del proyecto en dos semanas, aunque estos tiempos fueron cambiando según las necesidades.

Al finalizar las reuniones, se ha proporcionado a los desarrolladores una lista de tareas a realizar en un tiempo estipulado, de manera que se pueda verificar qué objetivos se han cumplido y cuáles no. Al empezar cada reunión se revisa las tareas realizadas en anteriores reuniones.

Para poder controlar todas estas tareas se ha hecho uso de la herramienta *Redmine*, mediante la cual es posible crear tareas y asignárselas al desarrollador que debe encargarse de realizarlas. En ella, los desarrolladores pueden indicar qué porcentaje han realizado de cada tarea e indicar el número de horas utilizado para ello. Gracias a esta

herramienta se puede contabilizar de una manera precisa las horas realizadas a lo largo de todo el proyecto.

## 1.4 Medios empleados

Para la realización de este proyecto se ha dispuesto de una serie de herramientas hardware y software.

El hardware necesario ha incluido:

- Ordenador portátil Apple MacBook Pro 13"
- Ordenador iMac (21.5 pulgadas, Procesador 2,5 GHz Intel Core i5, Memoria 8 GB, 1333 MHz DDR3 Gráficos AMD Radeon HD 6750M 512 MB)
- Teléfono móvil Apple iPhone 4S
- Escáner Canon CanoScan LiDE 700F
- Samsung Galaxy S i9000

En cuanto a las herramientas software se ha usado:

- MAC OS X Mountain Lion 10.6.8
- iOS 5
- Xcode 4.3.3
- Adobe Photoshop CS6 Extended Student and Teacher Edition
- Microsoft Office hogar y estudiantes 2010
- StartUML versión: 5.0.2.1570
- GanttProject 2.5.5 Praha
- OmniGraffle Professional

## 1.5 Terminología

En este apartado se recoge un glosario en el que se explican brevemente algunos de los términos utilizados en este documento con el fin de que el lector sea capaz de comprender el total del contenido.

### 1.5.1 Acrónimos y abreviaturas

DGT	- Dirección General de Tráfico
ICC	- International Color Consortium (Consortio Internacional del Color)
IDE	- Integrated Drive Electronics (Unidad Electrónica Integrada)
IOS	- Internet Operating System (Sistema Operativo de Internet)
OLE	- Object Linking and Embedding (Enlace e incrustación de objetos)
PDF	- Portable Document Format (Formato de Documento Portátil)

RACE	- Real Automóvil Club de España
RISC	- Reduced Instruction Set Computer (Computación de Juego de Instrucciones Reducidas)
TFG	- Trabajo Fin de Grado
JPG	- Joint Photographic Experts Group (Grupo de Fotógrafos expertos)
PNG	- Portable Network Graphics (Red de Gráficos Portátil)
HTML	- HyperText Markup Language (Lenguaje de Marcado de Hipertexto)
IRPF	- Impuesto sobre la Renta de las Personas Físicas
IVA	- Impuesto sobre el Valor Añadido

## 1.5.2 Definiciones

**App Store:** es un servicio para el iPhone y diversos dispositivos de Apple, creado por Apple Inc., que permite a los usuarios buscar y descargar aplicaciones informáticas.

**Seguridad vial:** consiste en la prevención de accidentes de tránsito o la minimización de sus efectos, especialmente para la vida y la salud de las personas, cuando tuviera lugar un hecho no deseado de tránsito. También se refiere a las tecnologías empleadas para dicho fin en cualquier vehículo de transporte terrestre

**Smartphone:** o teléfono inteligente es un teléfono móvil construido sobre una plataforma informática móvil, con una mayor capacidad de computación y conectividad que un teléfono móvil convencional.

**iPhone:** es una familia de teléfonos inteligentes multimedia con conexión a Internet, pantalla táctil capacitiva y escasos botones físicos diseñado por la compañía Apple Inc.

**iPad:** es un dispositivo electrónico tipo tableta desarrollado por Apple Inc. Las funciones son similares al resto de dispositivos portátiles de Apple, aunque la pantalla es más grande y su hardware más potente.

**Redmine:** es una herramienta para la gestión de proyectos que incluye un sistema de seguimiento de incidentes con seguimiento de errores. Otras herramientas que incluye son calendario de actividades, diagramas de Gantt para la representación visual de la línea del tiempo de los proyectos, wiki, foro, visor del repositorio de control de versiones, etcétera.

**Mac OS:** (Sistema Operativo de Macintosh) es el nombre del sistema operativo creado por Apple para su línea de computadoras Macintosh. Es conocido por haber sido el primer sistema dirigido al gran público en contar con una interfaz gráfica compuesta por la interacción del mouse con ventanas, Icono y menús.

**IOS:** es un sistema operativo móvil de Apple. Originalmente desarrollado para el iPhone. Apple, Inc. no permite la instalación de IOS en hardware de terceros.

**Microsoft Office:** es una suite de oficina que abarca e interrelaciona aplicaciones de escritorio, servidores y servicios para los sistemas operativos Microsoft Windows y Mac OS X.

**Microsoft Windows:** es el nombre de una familia de sistemas operativos desarrollados y vendidos por Microsoft.

**Microsoft Schedule plus:** fue un tiempo de gestión de producto de software desarrollado por Microsoft, pero fue suspendido como parte de Microsoft Office cuando la mayor parte de su funcionalidad fue incorporada en Outlook 97.

**Mapfre:** es un grupo español independiente, que desarrolla fundamentalmente actividades aseguradoras y reaseguradoras en 46 países.

**Android:** es un sistema operativo móvil basado en Linux, que junto con aplicaciones middleware está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas y otros dispositivos.<sup>9</sup> Es desarrollado por la Open Handset Alliance, la cual es liderada por Google.

**Google Play:** es una tienda de software en línea desarrollada por Google para los dispositivos Android. Es una aplicación que está preinstalada en la mayoría de los dispositivos Android y que permite a los usuarios buscar y descargar aplicaciones publicadas por desarrolladores terceros, alojada en Google Play.

**Middleware:** es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos. Éste simplifica el trabajo de los programadores en la compleja tarea de generar las conexiones que son necesarias en los sistemas distribuidos.

**Linux:** es un núcleo libre de sistema operativo basado en un sistema operativo portable, multitarea y multiusuario. Es uno de los principales ejemplos de software libre.

**Hardware:** corresponde a todas las partes tangibles de un sistema informático; sus componentes son: eléctricos, electrónicos, electromecánicos y mecánicos.

**Software:** al equipamiento lógico o soporte lógico de un sistema informático, comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas.

**Sistema Operativo:** es un programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes.

**Plugin:** (complemento) es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

**Game Center:** plataforma online de Apple dónde los jugadores pueden compartir estadísticas y progresos.

**NextStep:** es el sistema operativo orientado a objetos, multitarea que NeXT Computer, Inc. diseñó para ser ejecutados en los computadores NeXT

**Symbian:** es un sistema operativo que fue producto de la alianza de varias empresas de telefonía móvil. El objetivo de Symbian fue crear un sistema operativo para terminales móviles que pudiera competir con Android e IOS.

**Nokia:** es una empresa transnacional y es el segundo fabricante de teléfonos móviles del mundo, además de una de las principales empresas del sector de las telecomunicaciones.

**Xcode:** es el entorno de desarrollo integrado (IDE) de Apple Inc. y se suministra gratuitamente junto con Mac OS X. Xcode trabaja conjuntamente con Interface Builder, una herencia de NeXT, una herramienta gráfica para la creación de interfaces de usuario.

**Protocolo Bonjour:** es un protocolo para descubrir servicios en una red de área local.

**Mach-O:** es un formato del archivo que especifica el formato de archivos ejecutables.

**PowerPC:** es el nombre original de la arquitectura de computadoras de tipo RISC.

**RISC:** es un tipo de diseño de ordenadores generalmente utilizado en microprocesadores o microcontroladores con dos características fundamentales. Las instrucciones de tamaño fijo y presentado en un reducido número de formatos y sólo las instrucciones de carga y almacenamiento acceden a la memoria de datos.

**Intel:** es el mayor fabricante de circuitos integrados del mundo, según su cifra de negocio anual. La compañía estadounidense, es la creadora de la serie de procesadores x86, los procesadores más comúnmente encontrados en la mayoría de las computadoras personales.

**Bitmap:** es una estructura o fichero de datos que representa una rejilla rectangular de píxeles o puntos de color, denominada matriz, que se puede visualizar en un monitor, papel u otro dispositivo de representación.

**Objective-C:** es un lenguaje de programación orientado a objetos creado como un superconjunto de C.

**Scrum:** es un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.

## 1.6 Estructura de la memoria

Para facilitar la lectura de la memoria, se incluye a continuación un breve resumen de cada capítulo.

En el primer apartado: “Capítulo 1 Introducción”, se describen los motivos que han llevado a la elaboración de este proyecto, así como los objetivos que se desean alcanzar con la realización del proyecto. Además se incluye los medios utilizados para llevarlo a cabo y un glosario de términos y abreviaturas para facilitar la comprensión de la memoria.

En el segundo apartado: “Capítulo 2 Estado del arte” se realiza un estudio previo de las tecnologías existentes similares al objetivo del proyecto y a continuación se proporciona un razonamiento detallado del porqué se ha utilizado una determinada tecnología para la realización del proyecto y qué método de desarrollo se ha utilizado para llevar a cabo el proyecto.

En el tercer apartado: “Capítulo 3 Análisis” se definirá lo que se espera del sistema. Para ello se hará una descripción general del sistema y se concretará una lista de requisitos.

En el cuarto apartado: “Capítulo 4 Diseño” se realizará un estudio de los diseños realizados, por un lado el diseño externo, por medio de *paper prototypes* y se determinará cuál es el diseño final. Y por otro lado el diseño interno, que se dividirá en la creación de diagramas de secuencia, diagrama de clases, de imágenes y para terminar se hablará sobre la persistencia de los datos.

En el quinto apartado: “Capítulo 5 Plan de pruebas” se hará un plan de pruebas acorde con los requisitos definidos en el Capítulo 3.

En el sexto apartado: “Capítulo 6 Gestión del proyecto” se realizará una comparación de la planificación inicial del proyecto y la que realmente se ha llevado a cabo. Para finalizar se realizará un presupuesto de todo el proyecto.

En séptimo apartado: “Conclusión y trabajos futuros” se exponen las conclusiones finales a las que se ha llegado después de la realización del proyecto, haciendo una revisión de los objetivos iniciales y estudiando si se han cumplido debidamente cada uno de ellos. Además se realizará un breve comentario de lo aprendido durante este proyecto de un punto de vista personal. Finalmente se indican una serie de futuros trabajos que se pueden realizar como ampliación al desarrollo de este proyecto.

En octavo y último apartado: “Bibliografía” se indican las páginas y los documentos a los que se ha hecho algún tipo de referencia a lo largo del presente documento.

## Capítulo 2 Estado del arte

2.1 Criterios de búsqueda .....	24
2.2 Análisis de las aplicaciones relacionadas con los objetivos establecidos .....	25
2.2.1 Aplicaciones de Store de Apple .....	25
2.2.2 Aplicaciones de Google Play.....	28
2.2.3 Aplicaciones de internet .....	29
2.2.4 Conclusiones.....	32
2.3 Entorno tecnológico seleccionado .....	33
2.3.1 iPhone .....	34
2.3.2 IOS.....	35
2.3.3 Xcode.....	37
2.3.4 Adobe Photoshop.....	37
2.3.5 Microsoft Office .....	38
2.3.6 StarUML.....	39
2.3.7 GanttProject .....	40
2.3.8 Omnigraffle profesional .....	40
2.4 Método para realizar el proyecto: Desarrollo Ágil.....	41
2.4.1 The Agile Alliance .....	41
2.4.2 Scrum.....	42

El primer paso que se debe dar en cualquier proyecto basado en una aplicación es analizar las aplicaciones existentes en el mercado con el fin de buscar la similitud con nuestra aplicación o detectar carencias o errores para evitar y solucionar dichos problemas en la aplicación a desarrollar. También hay que tener en cuenta el contexto en el cual se va a desarrollar la aplicación.

En este capítulo se expone la información obtenida al investigar acerca de los temas que se tratarán a lo largo del desarrollo del proyecto. Para ello, se expone en primer lugar los criterios de búsqueda de las aplicaciones. Tras esto, un análisis de aplicaciones móviles y web relacionadas con los objetivos establecidos. Y para terminar se explica cuál es el entorno de desarrollo elegido y cuáles son sus características principales.

## 2.1 Criterios de búsqueda

El análisis se va a realizar mediante la búsqueda a través de distintas herramientas. Esta búsqueda se basa en aplicaciones de seguridad vial en todas las plataformas para dispositivos móviles y para la web. La búsqueda se ha centrado en seguridad vial para niños ya que es el objetivo principal de nuestra aplicación.

Las herramientas de búsqueda empleados han sido:

- Store de Apple, tanto en iPhone como en navegador web.<sup>1</sup>
- Google Play (Market de Android), tanto en un Smartphone (Samsung Galaxy S i9000) como en navegador web.<sup>2</sup>
- Buscador de Google.<sup>3</sup>

Los términos utilizados para la búsqueda en la web han sido los siguientes:

- “Seguridad vial”
- “Seguridad vial para niños”
- “Enseñar seguridad vial para niños”
- “App de seguridad vial”
- “Aprender seguridad vial niños”

Mientras que en las búsquedas en el iPhone y en el Smartphone en sus respectivos buscadores de aplicaciones han sido:

- “Seguridad vial”
- “Aprende seguridad vial”
- “Seguridad vial niños”

Estas búsquedas se han realizado en el mes de febrero de 2012.

El siguiente apartado se contemplarán las aplicaciones que se han aceptado como válidas, las demás aplicaciones encontradas han sido descartadas porque no se ajustaba a los criterios de búsqueda o a los objetivos de la aplicación, ya sea porque no era una aplicación centrada en la seguridad vial o bien no estaba orientado a los niños.



## 2.2 Análisis de las aplicaciones relacionadas con los objetivos establecidos

En este apartado se clasificarán las aplicaciones encontradas en las plataformas anteriormente indicadas.

Las aplicaciones encontradas en la Store de Apple han sido:

- Seguridad vial de RACE.
- Seguridad Vial de Confort Auto.
- Toca la señal, de la Fundación Mapfre.
- De dos en dos, de la fundación Mapfre.
- Un paseo seguro, de la fundación Mapfre.

Las aplicaciones encontradas en Google Play han sido:

- Seguridad vial de RACE.

Por último en Internet se han encontrado estos juegos o aplicaciones:

- Educapeques<sup>4</sup>
- Educación Skoda<sup>5</sup>
- Junta de Castilla y León<sup>6</sup>
- Educación vial de la DGT.<sup>7</sup>

A continuación se analizará a fondo cada una de las aplicaciones comentadas.

Para cada aplicación elegida se analizará su tecnología, cuál es su objetivo, a quien va dirigido, sus funcionalidades, sus puntos fuertes y débiles.

### 2.2.1 Aplicaciones de Store de Apple

Estas aplicaciones se dividen en dos grupos, ya que la aplicación de RACE se analizará en la parte de Android.

#### A. Confort Auto

Esta es la aplicación más simple que se ha encontrado en el mercado. Consiste en una aplicación orientada a los más pequeños para que aprendan seguridad vial.

Tienes dos apartados, uno en el que se les enseña las señales por medio de un cuestionario y otra que se le muestran dibujos didácticos, como un niño subido a una bici con el casco, y lo puede pintar por medio de una paleta de colores que se le muestra si pincha en un botón. Este apartado está realmente interesante ya que si el niño se cansa de leer puede ponerse a colorear. Además de ser un punto muy original dentro de una aplicación.

Los únicos puntos débiles que se le ha encontrado es que no hay un botón de atrás en la aplicación, el flujo por medio de la aplicación es realmente limitado. Además la paleta de colores únicamente hay seis y que el cuestionario puede cansar al niño y dejar de captar su atención.

Por lo demás es bastante completa y cumple con el objetivo, enseñar. La funcionalidad más importante es que el niño pueda realizar un juego o un test para jugar y aprender a la vez.



Ilustración 1 - Menús de confort auto

La conclusión de esta aplicación es que se debe de introducir un toque de originalidad a la aplicación y que no debe llevar mucho texto para cansar al usuario.

## B. Mapfre

En cuanto a las tres aplicaciones que nos ofrece Mapfre, son más orientadas a un juego para aprender. Este es el punto más fuerte que tiene que mientras estas aprendiendo las señales puedes estar divirtiéndote o dando un paseo en moto.

En “De dos en dos” es un juego que debes ir juntando señales que son iguales. Y su punto débil es que tampoco ofrece más información a parte de la señales de tráfico.

La funcionalidad que se puede extraer de estas aplicaciones es que mediante un sencillo juego se recuerda la importancia de la señales en la seguridad vial.

Ocurre lo mismo con “Toca las señales” es un juego con varios niveles donde tienes que agudizar el ingenio y la rapidez, es algo complicado para los más pequeños.

En cuanto a la última de las aplicaciones es un juego interactivo de cómo se debe respetar las normas de seguridad, esta aplicación esta orientadas más a jóvenes y no a niños.



Ilustración 2 - Mapfre juego

La conclusión de esta aplicación es que no solo debe basarse en un juego sino que debe tener distintos niveles de dificultad o distintos juego de aprendizaje dentro de nuestra aplicación.

## 2.2.2 Aplicaciones de Google Play

### A. RACE

Esta aplicación es la más completa que se ha encontrado en la búsqueda. Lo contempla prácticamente todo acerca de la seguridad vial. Está dirigida a todos los públicos, aunque es un poco seria y sin dinamismo para captar la atención de los más pequeños.

No obstante han hecho un gran trabajo en RACE para diseñar esta aplicación. La funcionalidad que se puede extraer de esta aplicación es una serie de menús para especificar lo que se quiere visualizar.



Ilustración 3 - RACE menús

Otra de las funcionalidades que se puede extraer de esta aplicación es que contiene una serie de videos explicativos.



Ilustración 4 - RACE videos

La conclusión de esta aplicación es que tiene muchísimas funcionalidades y es muy completa. Pero lo que se está buscando es originalidad y atracción al público más pequeño, esta aplicación no lo cumple.

Lo único a destacar es la introducción de videos en la aplicación. Esta posibilidad se ha contemplado para añadirla en la aplicación, pero finalmente se ha descartado por se cree que no encaja con los usuarios más pequeños.

## 2.2.3 Aplicaciones de internet

### A. Educapeques y Scoda

Este análisis se va hacer en conjunto ya que estas dos aplicaciones son muy similares. El objetivo de estas es enseñar seguridad vial a los más jóvenes.

El punto fuerte de estas aplicaciones es la funcionalidad que se puede extraer ellas, ya que es una de las más importantes y es que para los niños debe ir unida la diversión con el aprendizaje, se debe captar su atención con algún tipo de juego y con un diseño dinámico.

Su punto débil si es que tiene una excesiva carga de texto y esto para los niños más pequeños puede suponer una molestia o un problema.



Ilustración 5- Educapeques





Ilustración 6 - Scoda

En conclusión estas aplicaciones intentan captar la atención de los más pequeños mediante juegos pero en ciertas situaciones es excesiva la carga de texto produciendo al usuario una sensación de pesadez o aburrimiento.

## B. DGT

En esta web se puede encontrar distintos apartados, unidades didácticas y teoría por medio de documentos o bien juegos interactivos. En este análisis se va a centrar en los juegos interactivos que nos ofrece la dirección general de tráfico para la educación primaria de seguridad vial.

Estos juegos interactivos se dividen en tres: Juego de Foro, Lámina de errores y Lámina de trayectos.

En el primero se puede ver como por medio de una serie de preguntas vas avanzado en la acción que debes realizar. Además contabiliza el tiempo en que realizas cada acción y te van dando puntos según contestes bien las preguntas, y te resta puntos si las contestas mal.



En el tercero de los juegos es otra imagen pero esta vez tienes un personaje que interactúa con dicha imagen para recorrer distintos caminos. Es un poco complicado y no es intuitivo aunque las imágenes son bastante originales.



Ilustración 9 - DGT mapa

La conclusión de estos juegos interactivos es que se podría añadir un tiempo y una puntuación a los niveles en los que juegas. Además son sencillos y originales. El único inconveniente es que tienen demasiada carga de texto.

## 2.2.4 Conclusiones

Las conclusiones que se extraen al haber realizado el análisis de las aplicaciones similares que hay en el mercado es que si se trata de seguridad vial orientada a los niños el método para conseguir captar su atención es por medio de juegos sencillos y divertidos que les enseñen las normas a seguir, la señales y otros aspectos de la seguridad vial, de una forma dinámica.

Por este motivo se ha pensado en crear un juego interactivo con el usuario. Este juego puede ser la creación de láminas de errores dónde el usuario puede visualizar un situación real de su vida cotidiana y poder identificar las imprudencias o incidencias que se comenten. Con esto se pretende que el usuario pueda divertirse mientras está aprendiendo algunas de las normas en seguridad vial.

En el juego que se va a llevar a cabo es una mezcla de todos los puntos fuertes de las aplicaciones que se han visto y se pretende alejarse de los puntos débiles. El juego se centrará más en el comportamiento del usuario en una situación real, enseñando las



normas básicas de seguridad vial y un buen comportamiento. Contrario a centrarse únicamente en una sección de la seguridad vial, como puede ser la señales. Otro de los puntos fuerte que se ha adoptado es utilizar las capacidades de enseñanza de la DGT para la aplicación a realizar, ya que la DGT no tiene ninguna aplicación móvil de estas características, solamente está alojada en la web. Por último destacar que se ha intentado alejarse lo más posible de los texto extensos y hacer la aplicación lo más visual posible.

Después de explicar las características de cada una de las aplicaciones mencionadas, se realiza un análisis comparativo entre ellas teniendo en cuenta algunos puntos como son:

- La facilidad de captar la atención de los usuarios, es decir, si el usuario es capaz de estar visualizando la aplicación hasta que llegue a su objetivo.
- La sencillez de la aplicación, es decir, que no contenga gran cantidad de texto y que pueda resultar relativamente sencillo completar el juego.
- Lo divertido que puede ser para el usuario, es decir, si tras estar un rato con la aplicación no se cansa el usuario de utilizarla.
- La originalidad en cuanto a imágenes y textos, es decir, si le impresiona al usuario las imágenes que visualiza o si le es fácil leer los contenidos de texto.
- Si tiene diferentes niveles de dificultad.

El resultado de la comparación se puede consultar en la Tabla 2.1 que se muestra a continuación.

<b>Objetivos</b> <b>Aplicaciones</b>	<b>Captar</b> <b>la atención</b>	<b>Sencillo</b>	<b>Divertido</b>	<b>Original</b>	<b>Niveles de</b> <b>dificultad</b>
<b>Confort auto</b>	Bajo	Medio	Medio	Alto	No
<b>Mapfre</b>	Alto	Bajo	Medio	Bajo	Si
<b>RACE</b>	Bajo	Medio	Bajo	Bajo	No
<b>Educapeques</b> <b>y Scoda</b>	Alto	Bajo	Medio	Alto	No
<b>DGT</b>	Alto	Medio	Alto	Medio	No

Tabla 1- Comparación de aplicaciones

Todas estas aplicaciones han sido probadas con distintos usuarios para poder elaborar esta tabla con la mayor objetividad posible.

## 2.3 Entorno tecnológico seleccionado

En esta sección se exhiben cada una de las tecnologías empleadas en el presente proyecto, tanto hardware como software. Además se aportan las razones por las que se han escogido.

### 2.3.1 iPhone

iPhone es una familia de teléfonos inteligentes multimedia con conexión a Internet, pantalla táctil capacitiva y escasos botones físicos diseñado por la compañía Apple.

En junio de 2007, el iPhone original fue sacado al público por primera vez en Estados Unidos. Desde entonces, otras versiones mejoradas han salido a un ratio de aproximadamente una por año hasta llegar al último modelo, el iPhone 4S.

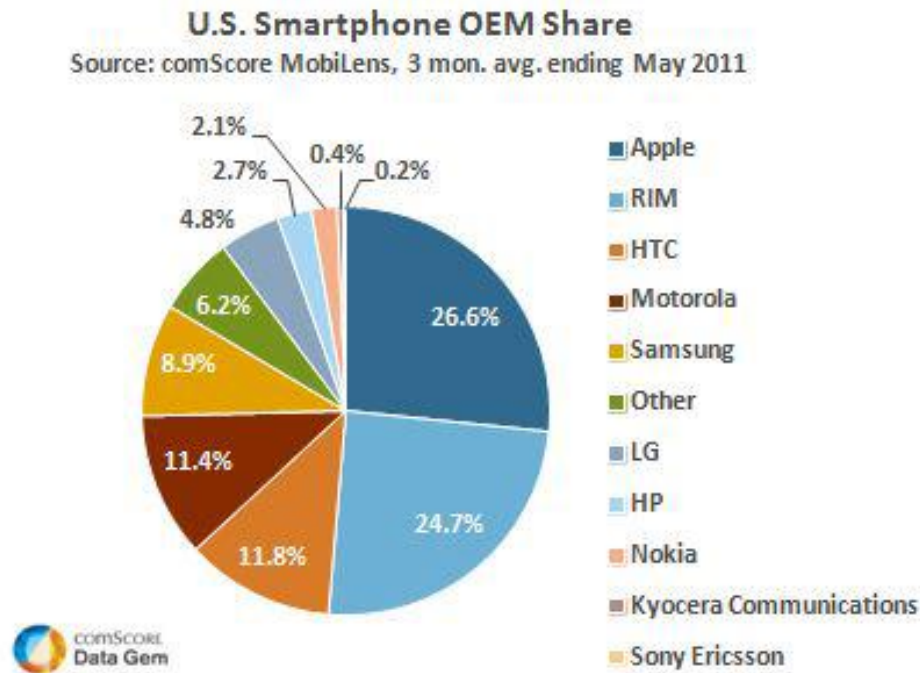


Ilustración 10- Comparativa del mercado de Smartphone por compañías.

Desde su aparición el iPhone se ha convertido en uno de los teléfonos más vendidos del mundo, habiendo llegado a la cifra de 108.624.000 unidades vendidas a nivel mundial hasta el 24 de marzo de 2011. Los datos indican que los teléfonos de Apple son los más vendidos de los teléfonos inteligentes en estados unidos, dato recogido en mayo de 2011.

En el segundo cuatrimestre de 2011 el iPhone tenía un 4.6% del mercado global de teléfonos móviles y un 18.2% del mercado global de Smartphone.

Una de las ventajas más destacadas que ofrece el dispositivo es la llamada App Store, un mercado de aplicaciones online para dispositivos Apple. La introducción del mercado App Store de Apple para el iPhone en julio de 2007 popularizó la distribución online de aplicaciones de terceros centradas en la plataforma. Siendo la App Store la que más ingresos generó en 2010.

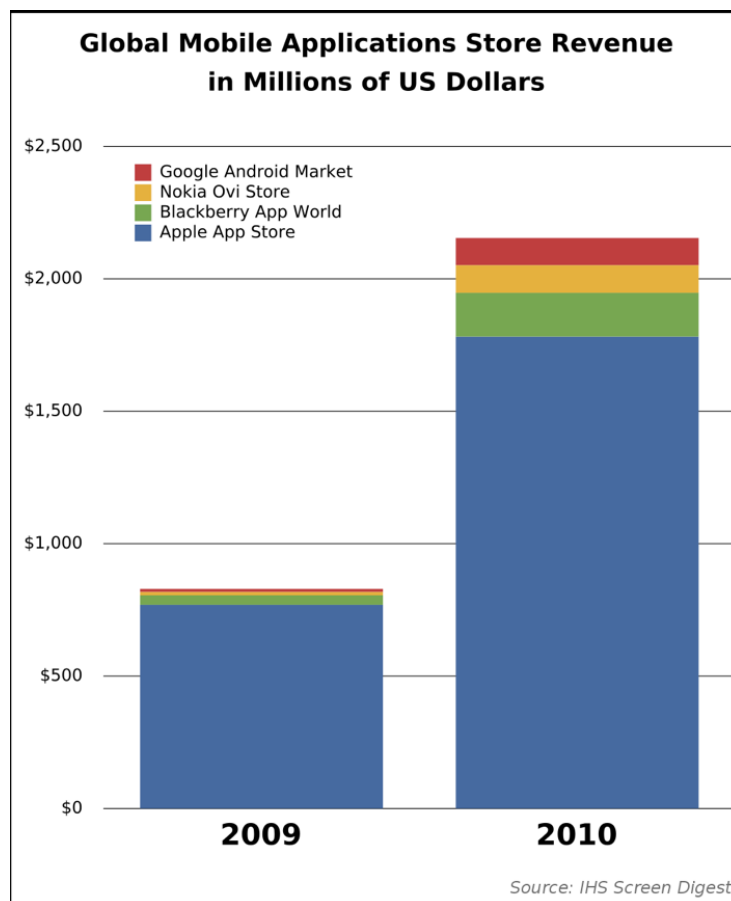


Ilustración 11 - Comparativa de ventas

Estos datos demuestran que el iPhone es uno de los dispositivos más activos en el mercado, teniendo casi 20% de las ventas de teléfonos inteligentes. Además la App Store ofrece un medio de ventas exitoso hasta la fecha, que es beneficioso para la distribución de la aplicación final de este proyecto.

En este proyecto se hará uso del iPhone 4S como terminal de pruebas ya que recoge todas las capacidades necesarias para el funcionamiento.

### 2.3.2 IOS

IOS, anteriormente conocido como iPhone OS, es el sistema operativo móvil de Apple. Por el 4 de Octubre de 2011 la App Store de Apple contiene más de 500,000 aplicaciones IOS, las cuales han sido colectivamente descargadas más de 18 billones de veces.



Ilustración 12 - Logotipo de IOS

Sus características más relevantes son:

- Programado en C, C++ y Objective-C.
- Multitáctil.
- Detección de movimientos del dispositivo: giroscopio y acelerómetro
- Multitarea (desde IOS 4.0).
- Pantalla de inicio simple: una única pantalla de inicio desde la que acceder a todas las aplicaciones.
- Organización en Ficheros.
- Centro de notificación: informa al usuario de los eventos que se producen en el sistema.
- Centro de juegos: plataforma online dónde los jugadores pueden compartir estadísticas y progresos, llamado Game Center.

El concepto IOS como sistema operativo es distinto respecto a otros sistemas operativos de escritorio. La entrada puede ser por medio de gestos táctiles como deslizamientos, pellizcos, giros, etc.

El sistema operativo es ampliamente utilizado por desarrolladores independientes y tiene una serie de librerías desarrolladas por Apple basadas en su antiguo NextStep, por lo que están creadas sobre una sólida base.

En el último cuatrimestre de 2010, IOS tenía un 26% de la cuota de teléfonos inteligentes en términos de unidades vendidas, detrás de Android de Google y Symbian de Nokia. La creación de la App Store, y su sencillo método de distribución, es una de las razones por las que el uso de este sistema operativo se ha extendido tan ampliamente entre los desarrolladores.

IOS es un sistema operativo con un sólido trasfondo y un amplio catálogo de librerías y documentación, elementos muy beneficiosos a la hora de afrontar la programación de un dispositivo iPhone.

### 2.3.3 Xcode

Xcode es una suite de herramientas desarrollado por Apple, para desarrollar software para Mac OS X y IOS. Xcode 4.3.3, la última versión, está disponible en la App Store para Mac de manera gratuita para desarrolladores registrados.



Ilustración 13 - Icono de Xcode

Entre las características de Xcode se encuentra la tecnología para distribuir la construcción del código fuente en varios ordenadores. Esta tecnología usa el protocolo Bonjour para descubrir automáticamente ordenadores que proveen servicios de compilador.

Gracias al formato ejecutable Mach-O, que permite binarios más grandes conteniendo el código para múltiples arquitecturas, Xcode puede construir binarios universales que permiten al software ejecutar tanto en PowerPC como en plataformas basadas en Intel.

Además los compiladores proporcionados con Xcode pueden compilar aplicaciones de 32 y 64 bits para ambas arquitecturas. Xcode también puede compilar aplicaciones para IOS que ejecutan sobre el procesador ARM.

Xcode además incluye el Interface Builder, un diseñador de interfaces que relaciona los objetos creados visualmente con el código fuente de una manera simple y rápida. De esta manera se evita tener que usar terceros programas para el diseño de interfaces ganando así tiempo en el desarrollo.

Xcode ofrece además herramientas de depuración y simuladores para iPhone e iPad, convirtiéndose por tanto en un IDE indispensable en el desarrollo de aplicaciones para IOS.

### 2.3.4 Adobe Photoshop

Es una aplicación informática basada en un taller de pintura y fotografía que trabaja sobre un "lienzo" y que está destinado para la edición, retoque fotográfico y pintura a base de imágenes de mapa de bits. La primera versión de photoshop se lanzó en febrero de 1990.



Ilustración 14 - Icono de Adobe photoshop

A medida que ha ido evolucionando el software ha incluido diversas mejoras fundamentales:

- La incorporación de un espacio de trabajo multicapa
- Inclusión de elementos vectoriales
- Gestión avanzada de color (ICC)
- Tratamiento extensivo de tipografías, control y retoque de color
- Efectos creativos
- Posibilidad de incorporar plugins de terceras compañías
- Exportación para sitios web entre otros

Photoshop se ha convertido en el estándar en retoque fotográfico, pero también se usa extensivamente en multitud de disciplinas del campo del diseño y fotografía, como diseño web, composición de imágenes bitmap, estilismo digital y edición.

Para este proyecto se ha utilizado Adobe Photoshop CS6 Extended Student and Teacher Edition para la realización de todas y cada una de las imágenes que se muestran en la aplicación.

### **2.3.5 Microsoft Office**

Microsoft Office es una suite de herramientas que abarca e interrelaciona aplicaciones de escritorio, servidores y servicios para los sistemas operativos Microsoft Windows y Mac OS X.

Fue lanzado por Microsoft en 1989 para Apple, más tarde seguido por una versión para Windows, en 1990.



La primera versión de Office contenía Microsoft Word, Microsoft Excel y Microsoft PowerPoint. Además, una versión "Pro" (profesional) de Office incluía Microsoft Access y Schedule Plus.

Con el tiempo, las aplicaciones de Office han crecido sustancialmente y de forma más estrecha con características compartidas:

- Un corrector ortográfico común
- La integración de datos OLE
- El lenguaje de secuencias de comandos de Microsoft
- Conversión a formato PDF

Para la realización de la memoria de este proyecto se ha llevado a cabo en la versión del 2010 de Microsoft office hogar y estudiantes.

### 2.3.6 StarUML

StarUML es una herramienta UML de código abierto con el objetivo de reemplazar grandes aplicaciones comerciales tales como *Rational Rose* y *Borland Together*.



Lo mejor de StarUML es que soporta la mayoría de los tipos de diagramas UML especificadas en la versión 2.0. Además una de sus características más importantes es que su código es compatible con C++ y Java.

Se puede comenzar a dibujar los gráficos manualmente o seleccionar las plantillas que contiene el archivo de instalación para modificarlas. Se ha elegido este programa por la sencillez y facilidad de realizar los diagramas.

### 2.3.7 GanttProject

GanttProject es un software de código abierto, para la planificación y gestión de proyectos. La última versión 2.0.7, disponible tanto para Windows, MacOSX o Linux. Entre sus múltiples funciones permite exportar el trabajo a distintos formatos como JPG, PNG, HTML, PDF o exportar/importar desde otros gestores de proyectos.



Ilustración 17 - Icono de GanttProject

### 2.3.8 Omnigraffle profesional

Omnigraffle es una aplicación de diagramas realizados por el Grupo Omni. Se construyó únicamente para Mac OS X y iPad.

Puede ser usado para crear diagramas de flujo, organigramas e ilustraciones. Cuenta con una interfaz de arrastrar y soltar. Además cuenta con plantillas de formas. Están disponibles como extensiones para Omnigraffle, y los usuarios pueden crear sus propias plantillas.



Ilustración 18 - Icono de Omnigraffle profesional



## 2.4 Método para realizar el proyecto: Desarrollo Ágil

Las Metodologías Ágiles constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de proyectos que las metodologías convencionales (ISO-9000, CMM, etc.) debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración.

Las metodologías de desarrollo ágiles son adaptativas en lugar de predictivas. Los métodos de ingeniería tienden a intentar planear una parte grande del proceso del software en gran detalle para un plazo largo de tiempo, esto funciona bien hasta que las cosas cambian. Así que su naturaleza es resistirse al cambio. Para estas metodologías, no obstante, el cambio es bienvenido.

Intentan ser procesos que se adaptan y crecen en el cambio, incluso al punto de cambiarse ellos mismos. Son orientadas a la gente y no orientados al proceso. Y su meta es definir un proceso que funcionará bien con cualquiera que lo use.

### 2.4.1 The Agile Alliance

En febrero de 2001, tras una reunión celebrada en Utah (EE.UU.), nace el término ágil aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.

Tras esta reunión se creó “The Agile Alliance”, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía ágil, es este manifiesto se valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

- Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

A continuación se muestran en forma de tabla las diferencias que existen entre las metodologías ágiles y las tradicionales:

Metodología Ágil	Metodología No Ágil (Tradicional)
Pocos artefactos	Más artefactos
Pocos roles	Más roles
No existe un contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo (además in-situ)	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes
Menos énfasis en la arquitectura	La arquitectura es esencial

Tabla 2- Comparación entre Metodología Ágil y Tradicional

Dentro de las metodologías ágiles una de las características principales es el uso de historias de usuario. Son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

## 2.4.2 Scrum

Por último se va a hacer referencia a la metodología ágil utilizada como base para la realización de este proyecto que es Scrum<sup>16</sup> y cómo se ha adaptado a nuestro entorno.

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión (ROI) para su empresa.

Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación.

Con Scrum, el cliente se entusiasma y se compromete con el proyecto dado que lo ve crecer iteración a iteración. Asimismo le permite en cualquier momento realinear el software con los objetivos de negocio, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración.

Esta metodología de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

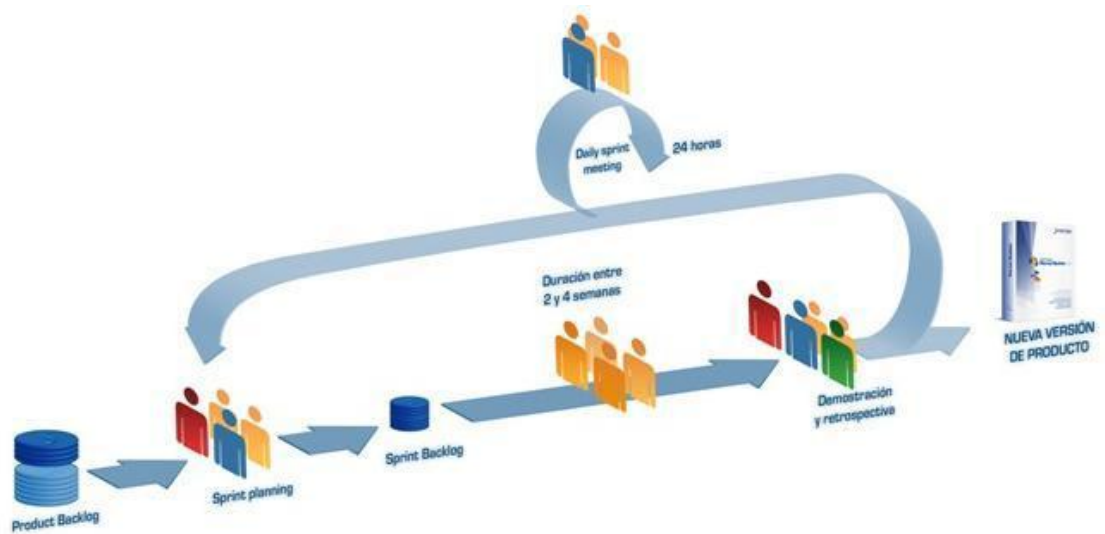


Ilustración 19 - Diagrama de flujo de Scrum

Los principales beneficios que proporciona la utilización de Scrum son los siguientes:

- Cumplimiento de expectativas: El cliente establece sus expectativas indicando el valor que le aporta cada requisito/historia del proyecto, el equipo los estima y con esta información establece su prioridad. De manera regular, en las demos de Sprint se comprueba que efectivamente los requisitos se han cumplido y se transmite feedback (retroalimentación) al equipo.
- Flexibilidad a cambios: Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado.
- Reducción del Time to Market: El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.
- Mayor calidad del software: La metodología de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.

- Mayor productividad: Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- Maximiza el ROI: Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.
- Predicciones de tiempos: Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está como trabajo pendiente.
- Reducción de riesgos: El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

# Capítulo 3 Análisis de requisitos

3.1 Descripción general .....	46
3.2 Requisitos de usuario.....	46
3.2.1 Requisitos de capacidad .....	47
3.2.2 Requisitos de restricción.....	48
3.3 Casos de uso .....	48
3.3.1 Diagrama .....	48
3.3.2 Descripción textual.....	49
3.4 Requisitos de software.....	50
3.4.1 Requisitos funcionales.....	51
3.4.2 Requisitos no funcionales.....	53

Una vez conocido el entorno tecnológico y el estado de la cuestión, se definirá clara y detalladamente lo que se espera del sistema. Es decir, las funcionalidades requeridas. Para ello se dará una descripción general del sistema como idea aproximada. Se concretará luego una lista de requisitos de usuario. A partir de los requisitos se generarán los casos de uso. Por último se establecerá, en base a lo anterior, la lista de requisitos software del sistema. En este caso al tratarse de una aplicación orientada a los usuarios más pequeños, los requisitos y casos de uso serán menores.

## 3.1 Descripción general

Como se ha comentado, el objetivo de este sistema es implementar una aplicación de entretenimiento y enseñanza de seguridad vial para los usuarios más pequeños en la plataforma IOS. Para implementar esta aplicación se ha decidido enfocarlo como si de un juego se tratase. Este juego consiste en seleccionar las imprudencias o incidencias que comenten los personajes en una imagen, o dicho de otra forma, en una lámina de errores debe descubrir cuáles son los errores o incidencias que se están cometiendo.

Las láminas están compuestas por distintos personajes realizando acciones. Estas pueden ser una imprudencia o puede ser que estén realizando un buen comportamiento. Según el usuario acierte dichas incidencias recibirá una puntuación u otra.

Las únicas funcionalidades que debe permitir la aplicación son:

- Acceder al menú principal
- Seleccionar en el menú el tipo de lámina, diferenciadas por la dificultad
- Poder visualizar un tutorial si el usuario lo desea
- Poder volver al menú principal
- Poder interactuar con la lámina de errores para poder llegar a superarla
- Poder visualizar la puntuación que ha recibido el usuario.

## 3.2 Requisitos de usuario

Los requisitos de usuario son los primeros a definir en el análisis y deben ser descritos por el cliente, que en este caso concreto es el propio alumno desarrollador de la aplicación. El patrón de especificación de requisitos es el siguiente:

Identificador	Nombre		
<b>Fuente</b>	De qué persona o personas proviene el requisito	<b>Prioridad</b>	La importancia que le da el cliente a la implementación de este requisito (Alta, media, baja)
<b>Descripción</b>	Descripción textual del requisito		

Tabla 3 - Plantilla de ejemplo de requisitos de usuario

Por cada requisito se utilizará un identificador único para cada uno de ellos con las siguientes abreviaturas:

- Requisito de capacidad → RUC
- Requisito de restricción → RUR

### 3.2.1 Requisitos de capacidad

RUC-01	El usuario selecciona una lámina		
Fuente	Cliente	Prioridad	Alta
Descripción	El usuario podrá seleccionar la lámina con la que jugar		

Tabla 4 - RUC-01 Seleccionar lámina

RUC-02	El usuario visualiza el tutorial		
Fuente	Cliente	Prioridad	Alta
Descripción	El usuario podrá visualizar un tutorial antes de jugar en cualquiera de las láminas		

Tabla 5 - RUC-02 Ver tutorial

RUC-03	El usuario interactúa con la lámina		
Fuente	Cliente	Prioridad	Alta
Descripción	El usuario podrá interactuar con la lámina una vez que empiece el juego		

Tabla 6 - RUC-03 Interactuar con la lámina

RUC-04	El usuario cambia de lámina		
Fuente	Cliente	Prioridad	Alta
Descripción	El usuario podrá cambiar la lámina que quiera seleccionar en el menú principal		

Tabla 7 - RUC-04 Cambiar lámina

RUC-05	El usuario puede volver al menú principal		
Fuente	Cliente	Prioridad	Alta
Descripción	El usuario podrá volver al menú principal antes de empezar el juego o antes de visualizar el tutorial		

Tabla 8 - RUC-05 Volver al menú principal.

RUC-06	El usuario puede visualizar su puntuación		
Fuente	Cliente	Prioridad	Alta
Descripción	El usuario podrá visualizar la puntuación obtenida al finalizar el juego, es decir, al encontrar todas las incidencias.		

Tabla 9 - RUC-06 Ver puntuación

### 3.2.2 Requisitos de restricción

<b>RUR-01</b>	<b>El sistema está en castellano</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema tendrá la totalidad de interfaz en castellano		

Tabla 10 - RUR-01 Idioma

<b>RUR-02</b>	<b>El sistema está en vertical</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Alta
<b>Descripción</b>	La aplicación estará orientada en vertical		

Tabla 11 - RUR-02 Sistema en vertical

<b>RUR-03</b>	<b>El sistema minimiza el tiempo de espera</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Baja
<b>Descripción</b>	El sistema responderá en menos de 7 segundos a todas las acciones realizadas dentro de la aplicación		

Tabla 12 - RUR-03 Minimizar tiempo de espera

<b>RUR-04</b>	<b>El sistema funciona en IOS 5</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Alta
<b>Descripción</b>	La aplicación funcionará en el sistema operativo IOS 5		

Tabla 13 - RUR-04 Funcionamiento

## 3.3 Casos de uso

Los casos de uso especifican qué hace el sistema desde el punto de vista del usuario. Es decir, describen un uso del sistema y cómo este interactúa con el usuario.

Con esta técnica podemos describir de una manera más clara lo que va a hacer el sistema. En este apartado se mostrará una descripción tanto visual como textual de los casos de uso del sistema. Cabe recordar que los casos de uso no serán muy numerosos debido al tipo de aplicación que se ha llevado a cabo.

### 3.3.1 Diagrama

En este diagrama se representa al posible actor, ya que en este caso solamente existirá un actor, y las interacciones que puede realizar con la aplicación.



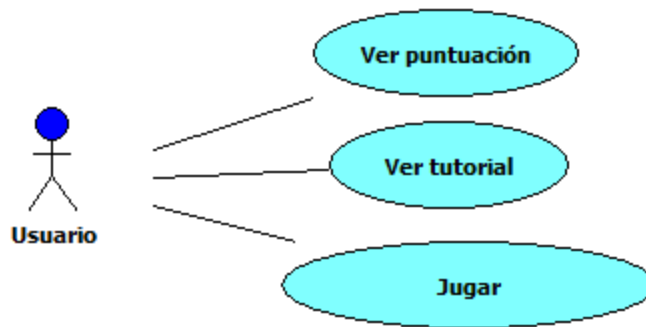


Ilustración 20 - Casos de uso

### 3.3.2 Descripción textual

Para realizar de una forma ordenada la descripción textual se utilizará una plantilla. La plantilla es la siguiente:

Identificador	Nombre
<b>Actores</b>	Lista de los actores involucrados en los casos de uso
<b>Dependencia</b>	Requisitos de usuario del que parte
<b>Descripción</b>	Objetivo del caso de uso y fuentes del requisito
<b>Suposiciones</b>	Condiciones que deben ser verdaderas para que el caso de uso concluya con éxito
<b>Flujo principal</b>	Interacciones entre actores y sistema que son necesarios para lograr el objetivo
<b>Flujo alternativo</b>	La excepciones al flujo principal serán consideradas en este apartado

Tabla 14- Plantilla de los casos de uso

La abreviatura para este tipo de identificador unívoco será CU.

CU-01	Ver puntuación
<b>Actores</b>	Usuario
<b>Dependencia</b>	RUC-06
<b>Descripción</b>	Visualizar la puntuación
<b>Suposiciones</b>	El usuario habrá pinchando en la última incidencia y así finalizará el juego
<b>Flujo principal</b>	1. Al usuario se le mostrará su puntuación nada más finalizar el juego
<b>Flujo alternativo</b>	No existe

Tabla 15- CU-01 Ver puntuación

<b>CU-02</b>	<b>Ver tutorial</b>
<b>Actores</b>	Usuario
<b>Dependencia</b>	RUC-02
<b>Descripción</b>	Visualizar el tutorial
<b>Suposiciones</b>	El usuario debe haber elegido una lámina de errores
<b>Flujo principal</b>	1. El usuario podrá elegir sí visualiza el tutorial
<b>Flujo alternativo</b>	1.1.Cancelar la opción de visualizar el tutorial, por lo que empezará el juego 1.2 Volver al menú principal

Tabla 16- CU-02 Ver tutorial

<b>CU-03</b>	<b>Jugar</b>
<b>Actores</b>	Usuario
<b>Dependencia</b>	RUC-01, RUC-03, RUC-04
<b>Descripción</b>	El usuario interactuará con la aplicación para finalizar el juego
<b>Suposiciones</b>	El usuario debe haber iniciado la aplicación.
<b>Flujo principal</b>	1. El usuario podrá elegir la lámina que se le muestra en pantalla. 1.1 El usuario podrá cambiar la lámina que se le muestre en pantalla. 2. El usuario cancelará el tutorial. 3. El usuario podrá seleccionar o elegir las incidencias mostradas en la lámina. 3.1 El usuario aceptará el mensaje mostrado por la aplicación 3.2 El usuario podrá seguir eligiendo incidencias hasta completarlas todas
<b>Flujo alternativo</b>	2.1 El usuario podrá elegir sí visualiza el tutorial 2.2 Volver al menú principal

Tabla 17- CU-03 Jugar

### 3.4 Requisitos de software

Una vez obtenidos los requisitos de usuario y los casos de uso se puede obtener la lista de los requisitos software, que serán la base fundamental de lo que va a ser finalmente la aplicación.

Estos requisitos se dividen en dos grupos fundamentales: los requisitos funcionales y los requisitos no funcionales. Se utilizará la misma plantilla que para los requisitos de usuario, lo único que les diferenciará será su identificador y el campo “fuente” que se sustituirá por dependencia. En este caso los requisitos funcionales será RSF y los no funcionales serán RSNF. El campo dependencia designará los requisitos de capacidad con los que comparte algún tipo de dependencia.

### 3.4.1 Requisitos funcionales

<b>RSF-01</b>	<b>El sistema muestra la imagen inicial</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar una imagen de carga inicial.		

Tabla 18 - RSF-01 Mostrar imagen inicial

<b>RSF-02</b>	<b>El sistema muestra el menú principal</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar el menú principal después de la imagen de carga inicial. Permitiendo al usuario la selección de una lámina de errores.		

Tabla 19 - RSF-02 Extensibilidad del funcionamiento

<b>RSF-03</b>	<b>El usuario puede cambiar de lámina al desplazar el dedo sobre la pantalla</b>		
<b>Dependencia</b>	RUC-04	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá permitir cambiar de lámina en el menú principal para que el usuario elija la lámina que desea jugar. Este cambio se hace cuando el usuario arrastrar el dedo por la pantalla.		

Tabla 20 - RSF-03 Cambio de lámina

<b>RSF-04</b>	<b>El usuario puede cambiar de lámina con un botón</b>		
<b>Dependencia</b>	RUC-04	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá permitir cambiar de lámina en el menú principal para que el usuario elija la lámina que desea jugar. Este cambio se hace cuando el usuario arrastrar el dedo por la pantalla. Este cambio se realiza cuando el usuario pincha en uno de los botones para cambiar de lámina.		

Tabla 21 - RSF-04 Cambio de lámina

<b>RSF-05</b>	<b>El usuario podrá seleccionar una lámina</b>		
<b>Dependencia</b>	RUC-01	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá permitir seleccionar una de las láminas del menú principal al usuario.		

Tabla 22 - RSF-04 Seleccionar lámina

<b>RSF-06</b>	<b>El usuario elegirá la opción de ver el tutorial</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar un menú para que el usuario pueda elegir visualizar el tutorial.		

Tabla 23 - RSF-06 Mostrar la opción tutorial

<b>RSF-07</b>	<b>El sistema mostrará el tutorial</b>		
<b>Dependencia</b>	RUC-02	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar el tutorial al usuario.		

Tabla 24 - RSF-07 Ver tutorial

<b>RSF-08</b>	<b>El usuario cancelará el tutorial</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar un menú para que el usuario pueda elegir cancelar el tutorial.		

Tabla 25 - RSF-08 Cancelar tutorial

<b>RSF-09</b>	<b>El sistema mostrará la lámina seleccionada por el usuario</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar la lámina una vez acabo el tutorial o cuando el usuario rechaza utilizar el tutorial.		

Tabla 26 - RSF-09 Mostrar lámina

<b>RSF-10</b>	<b>El usuario podrá seleccionar las incidencias</b>		
<b>Dependencia</b>	RUC-03	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá permitir al usuario seleccionar una incidencia nada más mostrar la lámina de errores.		

Tabla 27 - RSF-10 Seleccionar incidencias

<b>RSF-11</b>	<b>El sistema mostrará un mensaje de incidencia</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar un mensaje de incidencia cuando el usuario seleccione una incidencia. Este mensaje contendrá un información relevante a la incidencia, exponiendo si se ha acertado la incidencia o no.		

Tabla 28 - RSF-11 Mostrar un mensaje de incidencia

<b>RSF-12</b>	<b>El sistema mostrará el número de incidencias</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar en el mensaje de incidencias el número de incidencias restantes para finalizar el juego.		

Tabla 29 - RSF-12 Mostrar el número de incidencias

<b>RSF-13</b>	<b>El sistema mostrará imagen de felicitación</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema deberá mostrar una imagen de felicitación una vez se haya encontrado todas la incidencias en la lámina de errores.		

Tabla 30 - RSF-13 Mostrar imagen de felicitación

<b>RSF-14</b>	<b>El usuario podrá volver al menú principal</b>		
<b>Dependencia</b>	RUC-05	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema permitirá al usuario volver al menú principal para seleccionar una lámina. Esto sucede tanto al finalizar el juego como al elegir si visualizar o no el tutorial.		

Tabla 31 - RSF-14 Volver al menú principal

<b>RSF-15</b>	<b>El usuario podrá finalizar el tutorial</b>		
<b>Dependencia</b>		<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema permitirá al usuario terminar el tutorial y mostrará la lámina que ha seleccionado con anterioridad.		

Tabla 32 - RSF-15 Terminar tutorial

<b>RSF-16</b>	<b>El usuario podrá visualizar la puntuación</b>		
<b>Dependencia</b>	RUC-06	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema mostrará la puntuación obtenida a lo largo del juego, cuando este finalice. El juego finaliza cuando el usuario a encontrado todas las incidencias de la lámina.		

Tabla 33 - RSF-16 Ver puntuación

### 3.4.2 Requisitos no funcionales

Debido a que el cliente y el desarrollador son la misma persona en este caso, los requisitos no funcionales no darán más información que los requisitos de restricción y por lo tanto son los mismos. Ahora se volverán a adjuntar organizados por categorías.

#### A. Requisitos de interfaz

<b>RSNF-01</b>	<b>El sistema está en castellano</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Alta
<b>Descripción</b>	El sistema tendrá la totalidad de interfaz en castellano		

Tabla 34 - RSNF-01 Idioma

<b>RSNF -02</b>	<b>El sistema está en vertical</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Alta
<b>Descripción</b>	La aplicación estará orientada en vertical		

Tabla 35 - RSNF-02 Orientación.

## B. Requisitos de rendimiento

<b>RSNF -03</b>	<b>El sistema minimiza el tiempo de espera</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Baja
<b>Descripción</b>	El sistema responderá en menos de 7 segundos a todas las acciones realizadas dentro de la aplicación		

Tabla 36 - RSNF-03 Minimizar tiempo de espera

## C. Requisitos de operación

<b>RSNF -04</b>	<b>El sistema funciona en IOS 5</b>		
<b>Fuente</b>	Cliente	<b>Prioridad</b>	Alta
<b>Descripción</b>	La aplicación funcionará en el sistema operativo IOS 5		

Tabla 37 - RSNF-04 Funcionamiento

# Capítulo 4 Diseño

4.1 Diseño externo.....	56
4.1.1 Diseño inicial.....	56
4.1.2 Diseño final .....	60
4.2 Diseño interno .....	63
4.2.1 Diagrama de secuencia.....	63
4.2.2 Diagrama de clases .....	65
4.2.3 Creación de imágenes.....	67
4.2.4 Persistencia de datos.....	76

En este capítulo se presentan en primer lugar, los paper prototypes (prototipos en papel) modificados en cada una de las iteraciones en las que había cambios que afectaran de manera considerable a la interfaz de usuario. Estos paper prototypes son muy útiles para plasmar gráficamente los requisitos que se tienen para una aplicación, dándole a conocer al cliente un vistazo de cómo va a lucir la aplicación y pulir detalles antes de que se pase a una siguiente etapa de diseño o desarrollo.

A continuación se explican el diseño externo de la aplicación. Se explica las alternativas de diseño que se han barajado incluyendo sus ventajas e inconvenientes. Al final de esta sección se explica cuál ha sido la alternativa elegida y el porqué de la elección.

Posteriormente se presenta el diseño interno de la aplicación la cual enseñará con más detalle cómo va a ser dicha aplicación. Esta explicación se llevará a cabo con los diagramas de secuencias por cada caso de uso, os diagramas de secuencia necesarios para que un programador sea capaz de implementar la interacción de los objetos del sistema modelada en estos diagramas. A continuación un diagrama de clases con los métodos y atributos de cada clase, seguido con la creación de las imágenes por parte del desarrollador, y para terminar se explicará las decisiones en cuanto a la persistencia de datos.

## 4.1 Diseño externo

### 4.1.1 Diseño inicial

Este fue el primer diseño que se realizó. Este *paper prototype* se realizó en el mes de marzo del 2012.

Tras contemplar todas las posibilidades que podía realizar la aplicación se ha llegado a la conclusión de que se va a realizar una o varias historias interactivas en el que el usuario interacciona por medio de unas sencillas preguntas que podrá contestar por medio de imágenes. Si es correcta su respuesta avanzará en la historia que se le a propuesto si por el contrario su respuesta es errónea se le hará una pequeña sugerencia y podrá volver a responder la pregunta. Cuando haya alcanzado su objetivo o destino, el juego finalizará y se le recompensará con una grata felicitación y se dará la opción de volver al menú principal.

La herramienta que se ha utilizado para poder crear el prototipo ha sido el OmniGraffle Professional.

En primer lugar se diseñó el menú principal de la aplicación donde el usuario puede elegir a que historia interactiva quiere comenzar a jugar. En un principio se desarrollará una sola historia y si el tiempo y las condiciones son favorables se llevará acabo otra u otras historias.



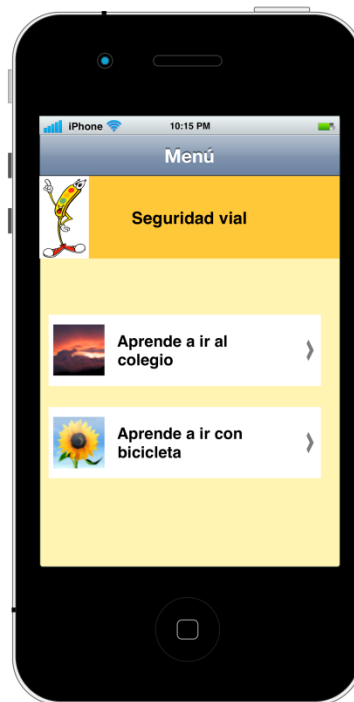


Ilustración 21 - Diseño inicial menú

Una vez seleccionado la historia o juego se le planteara al usuario el objetivo del juego y la situación en la que se encuentra, por medio de una imagen y posiblemente un audio explicativo de todo ello.

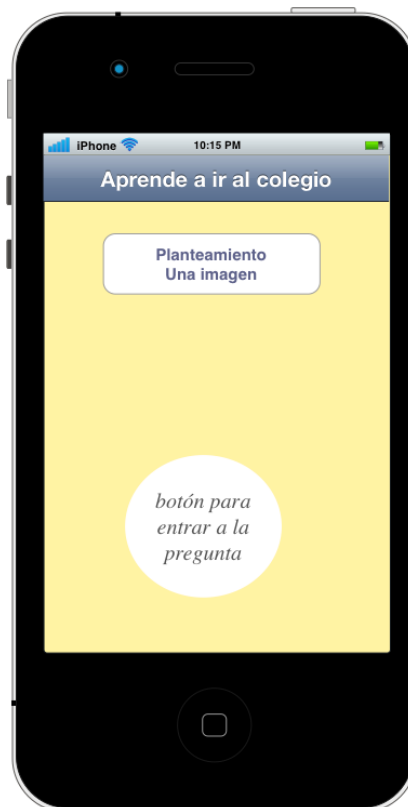


Ilustración 22 - Diseño inicial imagen

Una vez finalizado el audio el usuario podrá interactuar con dicha imagen para avanzar en la historia. Cuando seleccione uno de los “retos” a superar, le aparecerá una pregunta que podrá contestar pinchando de las respuestas que se le ofrece por medio de imágenes.

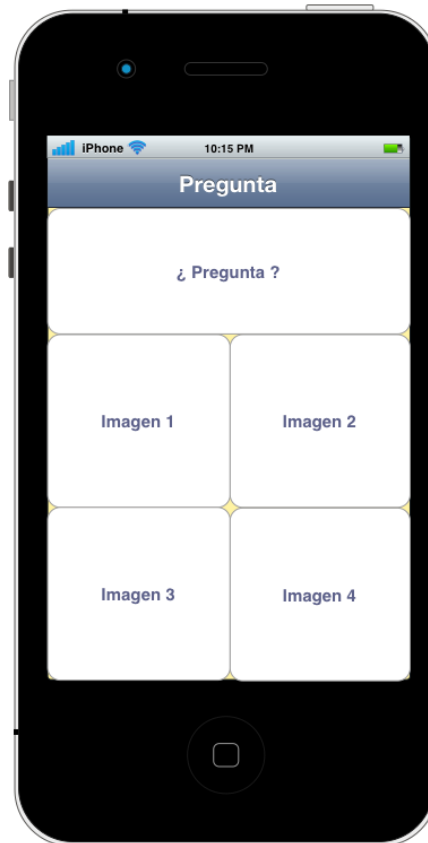


Ilustración 23 - Diseño inicial pregunta

Si el usuario acertará dicha pregunta volvería a la historia, sino la aplicación le proporcionará una sugerencia y le permitirá volver a contestar la pregunta.



Ilustración 24 - Diseño inicial error

Cuando el usuario finalice todas las preguntas propuestas y llegue a su destino, habrá finalizado el juego. Cuando esto ocurra la aplicación felicitará al usuario por su esfuerzo y le permitirá volver al menú principal para realizar si lo desea otro juego o historia.

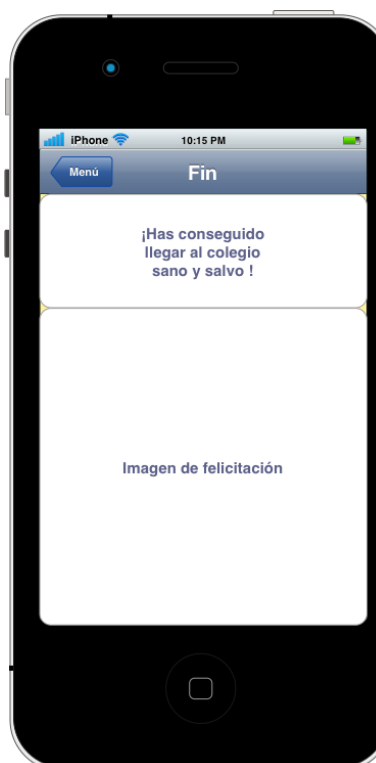


Ilustración 25 - Diseño interno felicitaciones

Esta fue la primera idea del diseño que se tuvo. Pero finalmente se descartó porque contenía mucho texto y con los usuarios que se probó se aburrían ya que no era capaz de captar la atención del usuario.

Por este motivo se pensó en hacer otro diseño totalmente distinto y más centrado en un juego, captando la atención y divirtiendo al usuario. Por ello se realizó el siguiente diseño.

Este nuevo diseño se llevó a cabo en el mes de mayo de 2012. Como con el anterior diseño no se captaba la atención se decidió pedir ayuda a Juan Carlos, uno de los profesores del departamento.

Él me sugirió centrarlo más en un juego, y dentro de este introducir de forma didáctica las normas básicas de seguridad vial. En un principio me pareció una gran idea poder desarrollar un videojuego dentro de la aplicación. Para ello se estudió los diferentes métodos para realizar un juego en la plataforma IOS<sup>14</sup>.

Tras varios días centrados en investigar los diferentes métodos se encontró el lenguaje de programación cocos2D<sup>13</sup>. Con dicho lenguaje se pueden crear juegos con gran cantidad de detalles y con todo tipo de funcionalidades.

Pero todo esto conllevaba empezar de cero con el aprendizaje ya que no tenía nada que ver con el lenguaje hasta hora estudiado Objective-C. Se fue estudiando este lenguaje y haciendo ciertos ejemplos sencillos pero ninguno de los ejemplos están enfocados a seguridad vial y tampoco al público más joven.

Por lo tanto después de intentar diseñar un juego, más complejo que el diseño anterior, se pensó que el lenguaje y el entorno tecnológico de dicha alternativa eran demasiados complejos.

Cabe destacar que durante varias semanas se intentó con mucho esfuerzo barajar dicha alternativa para llevarse a la implementación, pero finalmente se pensó que debía ser mejor en el lenguaje Objective-C.

### **4.1.2 Diseño final**

Este es el último diseño que se realizó y se llevó a cabo en Junio de 2012.

Como se ha dicho en el diseño anterior las desventajas que tenía era que no captaba la atención de los usuarios y contenía mucho texto produciendo cansancio y que el usuario dejase de utilizar dicha aplicación.

En este nuevo diseño se va a paliar dichos defectos y va conseguir resaltarlos.

Se adjuntan a continuación la imagen de menú que ahora si es atractiva para el usuario con colores llamativos e imágenes originales.



Ilustración 26 - Diseño final menú

Una vez que seleccione una lámina el usuario pasará a un menú para decidir si quiere visualizar el tutorial o bien prefiere pasar a la lámina para poder empezar a jugar.



Ilustración 27 - Diseño final tutorial

Ahora se va a mostrar un ejemplo de lámina para visualizar mucho mejor el diseño que se quiere conseguir.



Ilustración 28 - Diseño final lámina

Para finalizar el juego es necesaria una imagen de felicitación llamativa y que produzca al usuario una sensación de satisfacción.

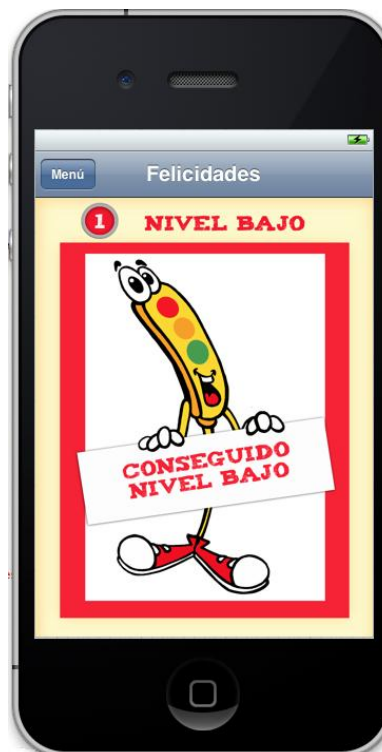


Ilustración 29 - Diseño final felicidades

Con este último y definitivo diseño se pretende cumplir con todos los objetivos de este proyecto. A continuación se llevará a cabo la explicación del diseño interno.

## 4.2 Diseño interno

### 4.2.1 Diagrama de secuencia

En este apartado se van a incluir ejemplos determinados de acciones que pueden llevar a cabo los usuarios en la aplicación a través de diagramas de secuencia. De esta forma se podrá visualizar claramente las relaciones que existen entre los usuarios y las diferentes partes de la aplicación. Para cada uno de los diagramas de secuencia se incluirá el diagrama en sí y una breve explicación sobre el desarrollo de la acción representada.

Cabe destacar que se ha realizado cada diagrama por cada caso de uso de la aplicación. Los diagramas muestran las clases por donde pasan para realizar el caso de uso y los métodos a utilizar.

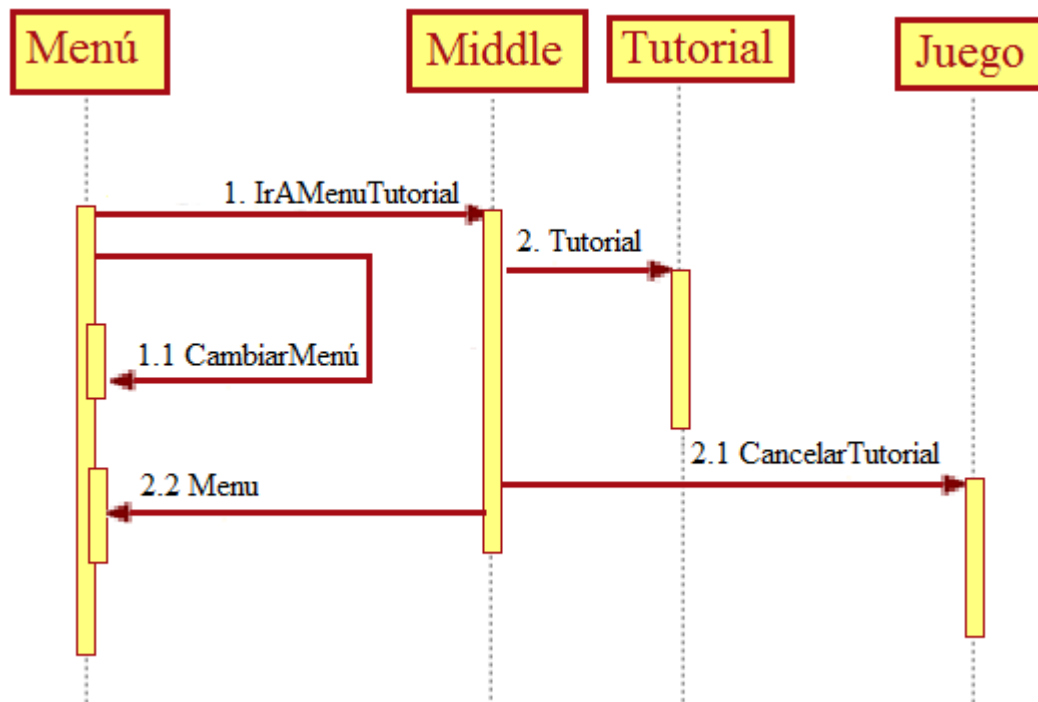


Ilustración 30 - Diagrama de Secuencia CU-02

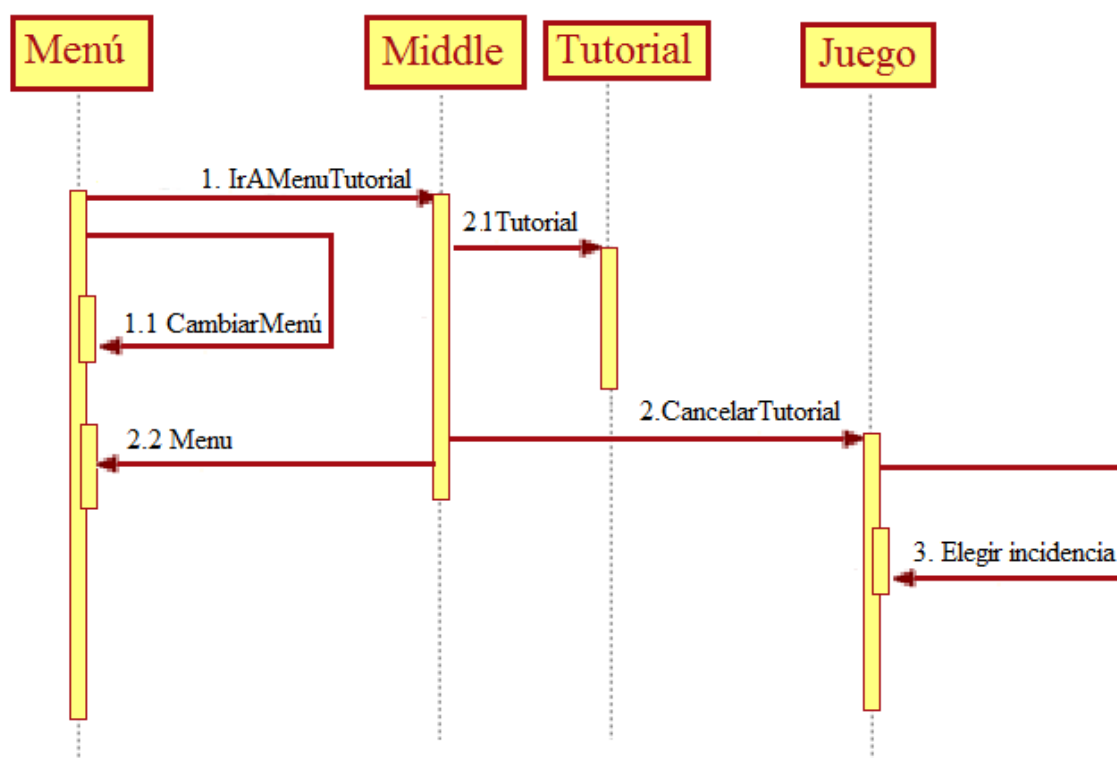


Ilustración 31 - Diagrama de Secuencia CU-03

A continuación se mostrará el diagrama de secuencia para el primer caso de uso que es el de poder visualizar la puntuación. Para poder obtener la puntuación se debe ejecutar anteriormente el caso de uso de Jugar.

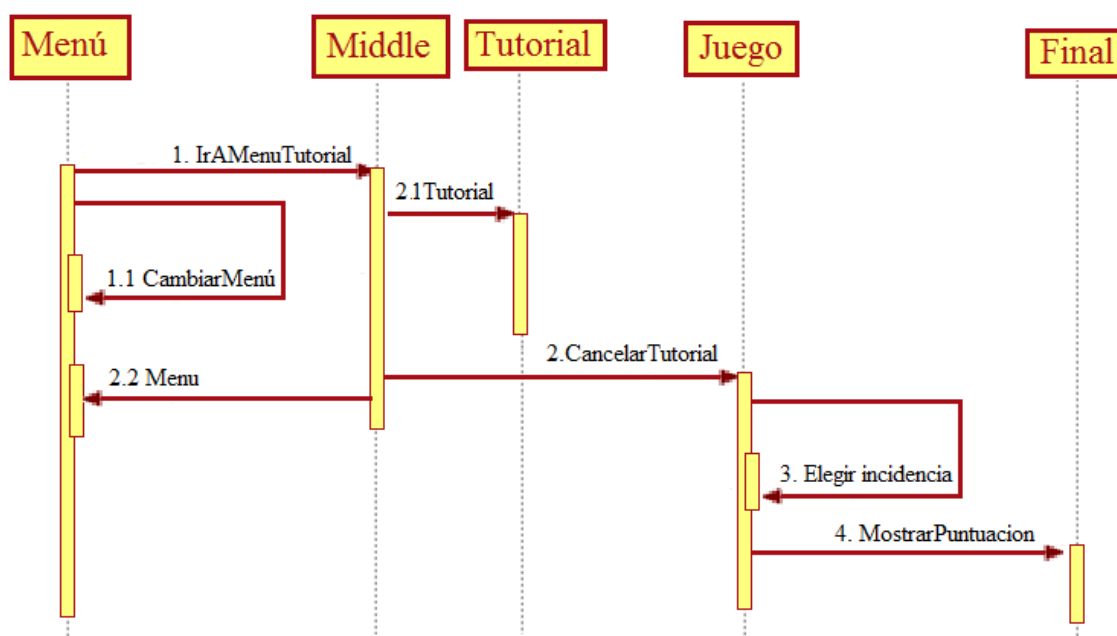


Ilustración 32 - Diagrama de secuencia CU-01



## 4.2.2 Diagrama de clases

Después de haber elegido la alternativa a desarrollar, se ha realizado el diseño de la aplicación por medio de diagramas. El diagrama de clases parte de la base del conjunto de interfaces, que se ve en el *paper prototypes*, y de los requisitos software.

Como se va a poder observar en la siguiente imagen hay cinco clases por cada lámina que se ha introducido. Esto ha sido una decisión del desarrollador para que en los posibles cambios que se realizaran en el proyecto se puedan incorporar de una manera más fácil otras láminas.

A continuación se muestra el diagrama de clases:

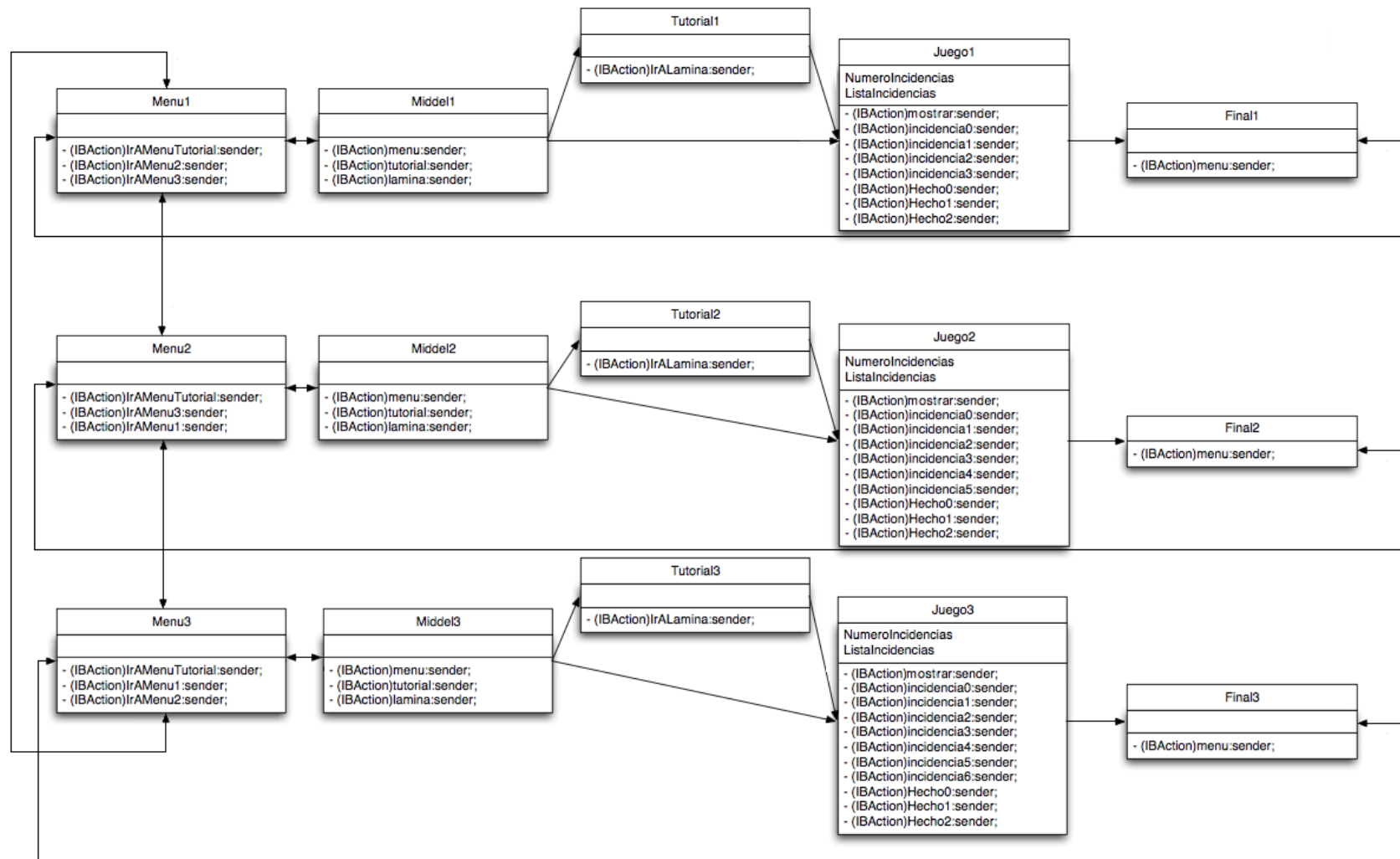


Ilustración 33 - Diagrama de clases

### 4.2.3 Creación de imágenes

A continuación se dará paso a la explicación detallada de la estética y realización del diseño de los personajes de dicha aplicación. Cabe destacar que este trabajo lleva una gran labor, tiempo y dedicación, ya que son muchos los factores a tener en cuenta.

En todo momento hay que tener consciencia del público al que va dirigido, ya que es un aspecto fundamental para decantarnos por un tipo de estética u otra.

En este caso la aplicación va dirigida a niños de temprana edad, por lo que la paleta de color no debe pasar desapercibida. Al diseñar para niños se suelen emplear colores llamativos, donde el contraste entre colores es lo que hace que no sean desapercibidos por ellos.

La cantidad de colores a utilizar y la tonalidad de los mismos va determinada por la edad del niño, es por ello que los juguetes y productos destinados a niños de edades más tempranas generalmente son con los colores primarios: rojo, amarillo y azul.

Para determinar la estética de la aplicación, a parte del color, se debe saber que los niños tienden a identificar fácilmente los elementos que los rodean, los símbolos son las primeras cosas que se aprenden (los animales, formas geométricas, etc.) Es por ello que funciona utilizar símbolos y elementos que puedan identificar. Sin embargo, es importante no olvidar la capacidad de imaginación que tienen los niños, por lo que el secreto de un diseño exitoso está en encontrar esa conexión entre la realidad e imaginación, permitiendo que los niños se identifiquen con la imagen gráfica y adquieran confianza y familiaridad con mayor rapidez.

Teniendo en cuenta todos estos factores, se decidió trabajar a partir de un semáforo como objeto de trabajo, ya que se compone de figuras geométricas muy sencillas, como son círculos, y con colores llamativos cerca de la paleta de primarios, en este caso prescindiendo del azul para dar prioridad al verde.

Todas las imágenes se han realizado exclusivamente para esta aplicación por medio de dibujos hechos en papel, escaneados y pasados al ordenador. Una vez allí, se colorean y se transforman gracias al programa Adobe Photoshop. Un buen ejemplo de este proceso se puede observar en la siguiente imagen, dónde se aprecian todos los aspectos anteriormente detallados, un elemento fácilmente reconocible, animado y compuesto a base de colores llamativos y figuras geométricas.

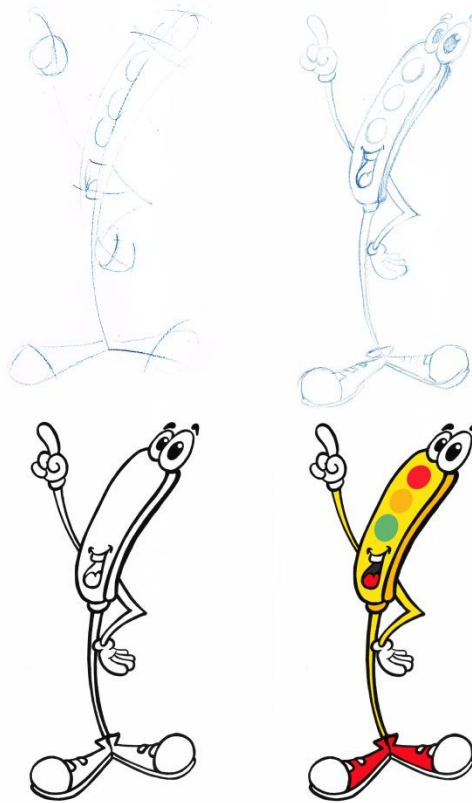


Ilustración 34 - Proceso de creación

Esta imagen será utilizada para crear la imagen de carga inicial de la aplicación.

La siguiente imagen creada ha sido la del icono de la aplicación. A partir de la idea del semáforo (ya que anteriormente se había creado un semáforo para la imagen de presentación o carga) y teniendo en cuenta la estética utilizada en las aplicaciones destinadas para iPhone, se decidió utilizar las formas geométricas y colores que más lo representan para componer un icono sintético y fácilmente identificable, a partir de los tres círculos: rojo, amarillo y verde.

La idea principal partía de componer una especie de semáforo abstracto representando un botón de entrada, con el fin del que el niño identifique que debe pulsarlo para entrar en dicha aplicación. Todo el icono está realizado con el programa Photoshop, creando los volúmenes y sombras característicos de los iconos de iPhone.

La imagen del resultado final del icono es la siguiente:



Ilustración 35 - Icono de la aplicación

Después de tener claros los diseños principales de la presentación de la aplicación, se pasa al diseño de las láminas de errores.

Para dar comienzo al diseño de las láminas, ha sido imprescindible la planificación de dicha lámina y de los elementos que debían aparecer en ellas, ya que los personajes que debían crearse debían responder a una serie de acciones propias de la educación vial y ciudadana.

Después de tener claro la cantidad de errores a descubrir y las personas que debían aparecer se comenzó a diseñar a los personajes. En la siguiente imagen se muestra los personajes que aparecen en la primera lámina de la aplicación.



Ilustración 36 - Personajes plantilla1

Los personajes han sido creados siguiendo los mismos pasos que el semáforo correspondiente a la imagen de entrada o carga de la aplicación ajuntado anteriormente. Dibujándolos manualmente, digitalizándolos a través del escáner y coloreándolos con el programa Photoshop.

A continuación se adjuntan también los personajes de las plantillas dos y tres, aunque estos personajes se realizaron después haber terminado por completo la plantilla uno.



Ilustración 37 - Personajes plantilla2



Ilustración 38 - Personajes plantilla3

Una vez que ya se tiene a los personajes se debe elaborar el paisaje de la lámina, es decir, el fondo. La creación del fondo se realizó exclusivamente con Photoshop, prescindiendo de la rotulación en negro que se observa en los personajes con el fin de que las acciones de los personajes destacaran de la escena o paisaje donde transcurre la acción.

Para poder crear este fondo se han utilizado muchas de las herramientas que nos proporciona Photoshop y a través del sistema de capas que le caracteriza, realizando los objetos a partir de figuras geométricas, sombreados de contorno y con tonalidades más suaves que las que aparecen en los personajes.

El resultado de la primera lámina es el siguiente.

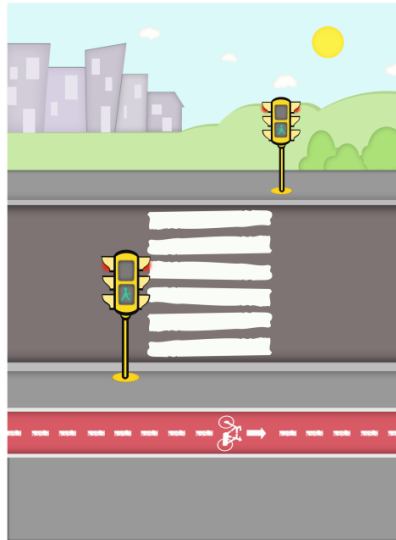


Ilustración 39 – Fondo plantilla1

Cómo en el apartado de diseño se decidió que cada lámina fuera más difícil. Cada lámina tiene más personajes y fondos más complejos. Aquí adjuntos los fondos de las láminas dos y tres.

Todos a partir de una misma estética y con la misma gama de colores.



Ilustración 40 - Fondo plantilla2

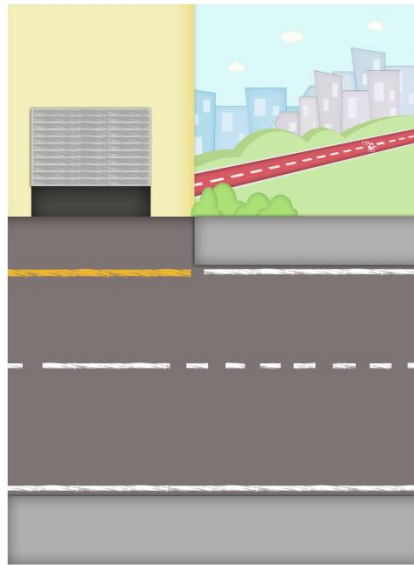


Ilustración 41 - Fondo plantilla3

Finalmente, una vez que ya se tienen los personajes y los fondos, se compone la escena por medio de la función de multicapas de Photoshop, una función muy útil para poder transportar de un lado a otro las diferentes escenas y personajes, hasta llegar a la composición final deseada.

Posteriormente adjunto el resultado final de las tres láminas completas para poder visualizar el trabajo realizado.

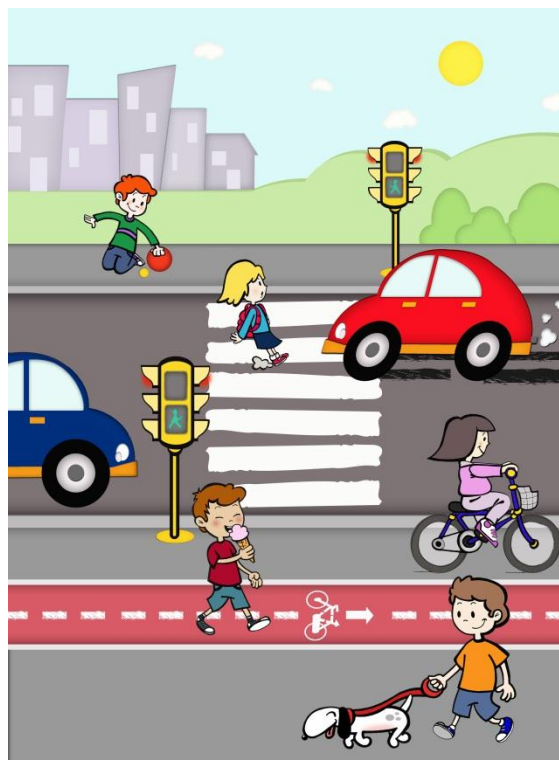


Ilustración 42 - Plantilla1



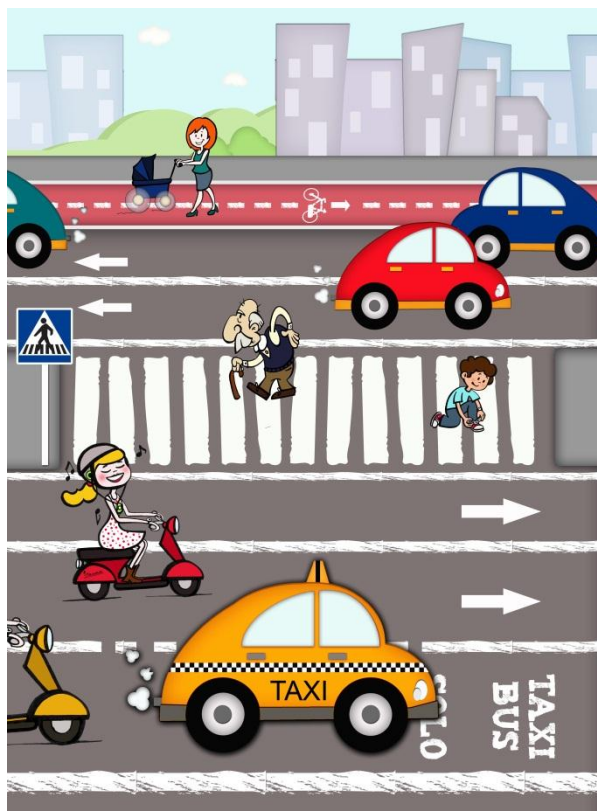


Ilustración 43 - Plantilla2



Ilustración 44 - Plantilla3





Ilustración 46 - Felicitaciones 1



Ilustración 47 - Felicitaciones 2

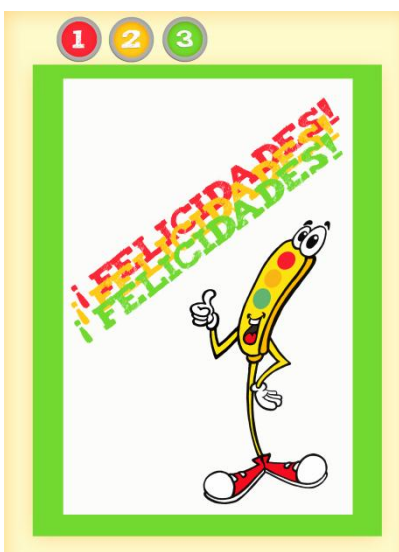


Ilustración 48 - Felicitaciones 3

## 4.2.4 Persistencia de datos

En este apartado se va a comentar las decisiones en cuanto a los datos se refiere.

El primer tema a abordar es la base de datos. En este caso se ha decidido que no exista como tal, una base de datos para almacenar las láminas de la aplicación sino que este en local, alojado en el dispositivo móvil. Esta decisión es debida a que en este proyecto se ha centrado más en la interacción con el usuario que en el sistema de almacenar los datos.

Otra posibilidad que se planteó hacer era una pequeña base de datos local para que la información estuviera protegida y que fuera más fácil la creación de nuevas láminas. Pero por el mismo motivo que el anterior se decidió no realizarlo porque se escapaba de los objetivos de la aplicación.

Cabe destacar que una opción para que la aplicación fuera más próspera sería la creación de una base de datos remota. Esto permitiría poder alojar láminas después de la adquisición de la aplicación. Esta alternativa proporcionaría ganancias ya que se podría descargar la aplicación gratuita pero si se desea por parte de los usuario tener más láminas con las que jugar poder descargarlas por un módico precio.

Es realmente interesante esta última alternativa a la persistencia de datos pero como se ha reiterado a lo largo de este capítulo, queda fuera del alcance de este proyecto, pero se propondrá para las posibles mejoras futuras que pueda tener la aplicación.

# Capítulo 5 Plan de pruebas

5.1 Pruebas de aceptación.....	78
5.1.1 Resultado de la pruebas .....	82

Las Pruebas de Software, son una serie de procesos de evaluación del software cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto al cliente. Las pruebas de software son una actividad más en el proceso de calidad.

A continuación se va a especificar un plan de pruebas de aceptación válidas para verificar todos los requisitos establecidos. Todas ellas deben ser superadas para considerar que el proyecto ha finalizado con éxito cumpliendo todos los objetivos establecidos. Finalmente se mostrará una matriz que representará la relación entre los requisitos y las pruebas que verifican su funcionalidad.

## 5.1 Pruebas de aceptación

El patrón de especificación de las pruebas es el siguiente:

Identificador	Nombre
<b>Entrada esperada</b>	Dato o acción de entrada que debe proporcionarse para poder llevar a cabo la prueba
<b>Salida esperada</b>	Resultado de salida que debe obtenerse una vez realizada la prueba
<b>Dependencia</b>	Dependencia existente entre la prueba y los casos de uso.

Tabla 38 - Plantilla de Prueba de aceptación

El identificador será unívoco y su abreviatura será PA.

PA-01	El usuario abre la aplicación
<b>Entrada esperada</b>	El usuario abre la aplicación.
<b>Salida esperada</b>	Se muestra la imagen de carga inicial y tras varios segundos con un máximo de 7 segundos, desaparece dicha imagen y se carga el menú principal.
<b>Dependencia</b>	

Tabla 39 – PA-01 Abrir la aplicación

PA-02	El usuario se desplaza por el menú
<b>Entrada esperada</b>	El usuario arrastra el dedo sobre el menú principal.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra la siguiente lámina de errores. Si el usuario se encuentra en la primera lámina le cambia de vista y va hasta la segunda. Si bien, está en la segunda cambia a la tercera y si está en la tercera vuelve a la primera.
<b>Dependencia</b>	CU-03

Tabla 40 – PA-02 Desplazarse en el menú

<b>PA-03</b>	<b>El usuario cambia de lámina</b>
<b>Entrada esperada</b>	El usuario pincha en botón de una lámina distinta.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra la lámina seleccionada. Si se encuentra en la primera lámina y pincha sobre el botón de la primera lámina el sistema no hará nada. Esto ocurre en las demás láminas, si se encuentra en la segunda y pincha el botón de la segunda tampoco hará nada y con la tercera lámina y el tercer botón ocurrirá lo mismo.
<b>Dependencia</b>	CU-03

Tabla 41 – PA-03 Cambiar de lámina

<b>PA-04</b>	<b>El usuario selecciono una lámina</b>
<b>Entrada esperada</b>	El usuario pincha dentro de la imagen de la lámina.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra un menú donde el usuario puede acceder al tutorial si lo desea o por el contrario cancelarlo. Además tiene la opción de volver al menú principal. Este menú consta de tres botones: Ver tutorial, Ya sé jugar y Menú
<b>Dependencia</b>	CU-03

Tabla 42 – PA-04 Seleccionar lámina

<b>PA-05</b>	<b>El usuario permite ver el tutorial</b>
<b>Entrada esperada</b>	El usuario selecciona la opción de “tutorial”.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra una imagen en la cual se explica al usuario cómo puede encontrar las incidencias en una lámina de errores.
<b>Dependencia</b>	CU-02

Tabla 43 – PA-05 Ver tutorial

<b>PA-06</b>	<b>El usuario cancela el tutorial</b>
<b>Entrada esperada</b>	El usuario selecciona la opción de “Ya sé jugar”.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra la imagen que corresponde a la lámina de errores seleccionada con anterioridad. Nada más mostrar dicha imagen se observa un mensaje de alarma dando la bienvenida al usuario a dicha lámina.
<b>Dependencia</b>	CU-03

Tabla 44 – PA-06 Cancelar tutorial

<b>PA-07</b>	<b>El usuario vuelve al menú principal desde el menú tutorial</b>
<b>Entrada esperada</b>	El usuario selecciona “Menú”.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra de nuevo el menú principal. Si el usuario ha entrado a dicho tutorial por la segunda lámina, al volver al menú principal se volverá a la selección de la segunda lámina. Esto mismo ocurre con la primera y la tercera lámina.
<b>Dependencia</b>	

Tabla 45 – PA-07 Volver al menú principal Tutorial

<b>PA-08</b>	<b>El usuario selecciona una incidencia</b>
<b>Entrada esperada</b>	El usuario pincha sobre un personaje que aparece en la imagen de la lámina de errores.
<b>Salida esperada</b>	El sistema lanza un mensaje que informa al usuario si se trata de una incidencia o no. Además muestra porqué es una incidencia o no y te muestra las incidencias que le quedan al usuario para terminar la lámina. En caso de que si fuese una incidencia, el número de incidencias mostrado sería uno menos ya que se ha acertado. Además si el usuario vuelve a pinchar en una de las incidencias el sistema le muestra un mensaje recordándole que ya ha acertado dicha incidencia.
<b>Dependencia</b>	CU-03

Tabla 46 – PA-08 Seleccionar incidencia

<b>PA-09</b>	<b>El usuario acepta un mensaje de incidencia</b>
<b>Entrada esperada</b>	El usuario pincha en el mensaje de incidencia para seguir jugando con la lámina.
<b>Salida esperada</b>	El sistema tras haber lanzado el mensaje le permite al usuario por medio de un botón que puedan continuar con el juego, volviendo a la lámina.
<b>Dependencia</b>	CU-03

Tabla 47 – PA-09 Aceptar mensaje

<b>PA-10</b>	<b>El sistema le muestra la felicitación</b>
<b>Entrada esperada</b>	El usuario pincha sobre un mensaje de incidencias mostrado por el sistema y este mensaje es el último para que el número de incidencias llegue a cero.
<b>Salida esperada</b>	El sistema nada más que acepte este mensaje, cambiara de vista a una imagen de felicitación y mostrará un mensaje que muestre la felicitaciones de una manera textual.
<b>Dependencia</b>	CU-03

Tabla 48 – PA-10 Felicitaciones

<b>PA-011</b>	<b>El usuario vuelve al menú principal desde el menú felicitaciones</b>
<b>Entrada esperada</b>	El usuario selecciona “volver al menú”, para posteriormente seguir jugando con otras láminas.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra de nuevo el menú principal, a la primera de la láminas.
<b>Dependencia</b>	

Tabla 49 – PA-11 Volver al menú principal Felicitaciones



<b>PA-12</b>	<b>El usuario termina el tutorial</b>
<b>Entrada esperada</b>	El usuario selecciona la opción de “terminar”.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra la imagen que corresponde a la lámina de errores seleccionada con anterioridad. Nada más mostrar dicha imagen se observa un mensaje de alarma dando la bienvenida al usuario a dicha lámina.
<b>Dependencia</b>	CU-02

Tabla 50 – PA-05 Terminar tutorial

<b>PA-13</b>	<b>El usuario visualiza la puntuación</b>
<b>Entrada esperada</b>	El usuario selecciona la opción de “terminar”.
<b>Salida esperada</b>	El sistema cambia de vista y le muestra la imagen que corresponde a la lámina de errores seleccionada con anterioridad. Nada más mostrar dicha imagen se observa un mensaje de alarma dando la bienvenida al usuario a dicha lámina.
<b>Dependencia</b>	CU-01

Tabla 51 - PA-13 Ver puntuación

A continuación se mostrará una matriz que representará la relación entre los requisitos de software funcionales y las pruebas que verifican su funcionalidad, para que se asegure que todos los requisitos están recogidos en las pruebas realizadas.

	PA-01	PA-02	PA-03	PA-04	PA-05	PA-06	PA-07	PA-08	PA-09	PA-10	PA-11	PA-12	PA-13
RSF-01	X												
RSF-02	X												
RSF-03		X											
RSF-04			X										
RSF-05				X									
RSF-06				X									
RSF-07					X								
RSF-08						X							
RSF-09						X							
RSF-10								X					
RSF-11								X	X				
RSF-12								X					
RSF-13										X			
RSF-14							X				X		
RSF-15												X	
RSF-16													X

Tabla 52 - Matriz RSF x PA

### 5.1.1 Resultado de la pruebas

Una vez finalizado el desarrollo de la aplicación y pasado la batería de pruebas se observa que los resultados han sido satisfactorios, la ejecución de todas las pruebas se ha llevado a cabo de forma correcta. La tabla que representa los resultados obtenidos en dichas pruebas es la siguiente:

Prueba	Resultado
PA-01	Correcto
PA-02	Correcto
PA-03	Correcto
PA-04	Correcto
PA-05	Correcto
PA-06	Correcto
PA-07	Correcto
PA-08	Correcto
PA-09	Correcto
PA-10	Correcto
PA-11	Correcto
PA-12	Correcto
PA-13	Correcto

Tabla 53 - Resultado de las pruebas

# Capítulo 6 Gestión del proyecto

6.1 Planificación del proyecto .....	85
6.1.1 Planificación inicial .....	85
6.1.2 Planificación final.....	87
6.2 Análisis económico .....	89
6.2.1 Coste personal .....	89
6.2.2 Coste Hardware .....	89
6.2.3 Coste Software.....	90
6.2.4 Costes indirectos.....	91
6.2.5 Costes totales .....	92

En este apartado se pretende explicar detalladamente todos los aspectos que forman la gestión del proyecto. Para ello se van a distinguir dos puntos claramente diferenciados: Primero, se van a explicar cada una de las fases planificadas por las que ha pasado el proyecto, y se realizará una comparación analizando la planificación inicial y la que realmente se ha llevado a cabo; en segundo lugar, se va a realizar un análisis exhaustivo donde se verá reflejado el coste que ha supuesto el desarrollo de la aplicación y los posibles beneficios obtenidos si esta se introdujera en el mercado.

## 6.1 Planificación del proyecto

En este primer apartado se detallarán todas las tareas realizadas a lo largo del proyecto y se realizará una comparación entre la planificación inicial y final del mismo tomando como referencia el diagrama de Gantt elaborado en la primera fase.

### 6.1.1 Planificación inicial

En primer lugar se encuentra la tarea de planificación. En ella se realizó la estimación de los días de trabajo que iban a transcurrir con la realización de cada tarea hasta la fecha límite (Dicha fecha se ha considerado el 6 de Agosto). Cabe destacar que es en este primer bloque de tareas cuando se realizó el diagrama de Gantt mostrado posteriormente.

Más tarde daría comienzo el conjunto de tareas relacionadas con el análisis. En esta parte se planificaron las tareas de análisis de aplicaciones similares a la desarrollada (esto del arte), que estaban encaminadas a proporcionar la información necesaria para conocer si el desarrollo de la aplicación supondría algo innovador y útil en cuanto a las aplicaciones ya existentes se refiere. De la misma forma, se planificaron las tareas relacionadas con el análisis de alternativas y, por último, la obtención de requisitos.

Posteriormente, da comienzo la fase de diseño de la aplicación. En dicha fase se planifica analizar la arquitectura del sistema, identificando el tipo de aplicación a desarrollar y los componentes que la forman.

Mientras el proceso de diseño es llevado a cabo, se planifica implementar y codificar la aplicación. Se planificó en primer lugar la instalación del entorno de programación y el respectivo aprendizaje del lenguaje a utilizar.

Finalmente, se realizó la estimación de la fase de pruebas y la comprobación de los últimos errores y retoques. Todos los conjuntos de tareas descritos anteriormente serán realizados de forma paralela a la presente documentación, completando los diferentes apartados del documento desde el primer día de trabajo.

El diagrama de Gantt elaborado que refleja la planificación de las tareas mencionadas es el siguiente:

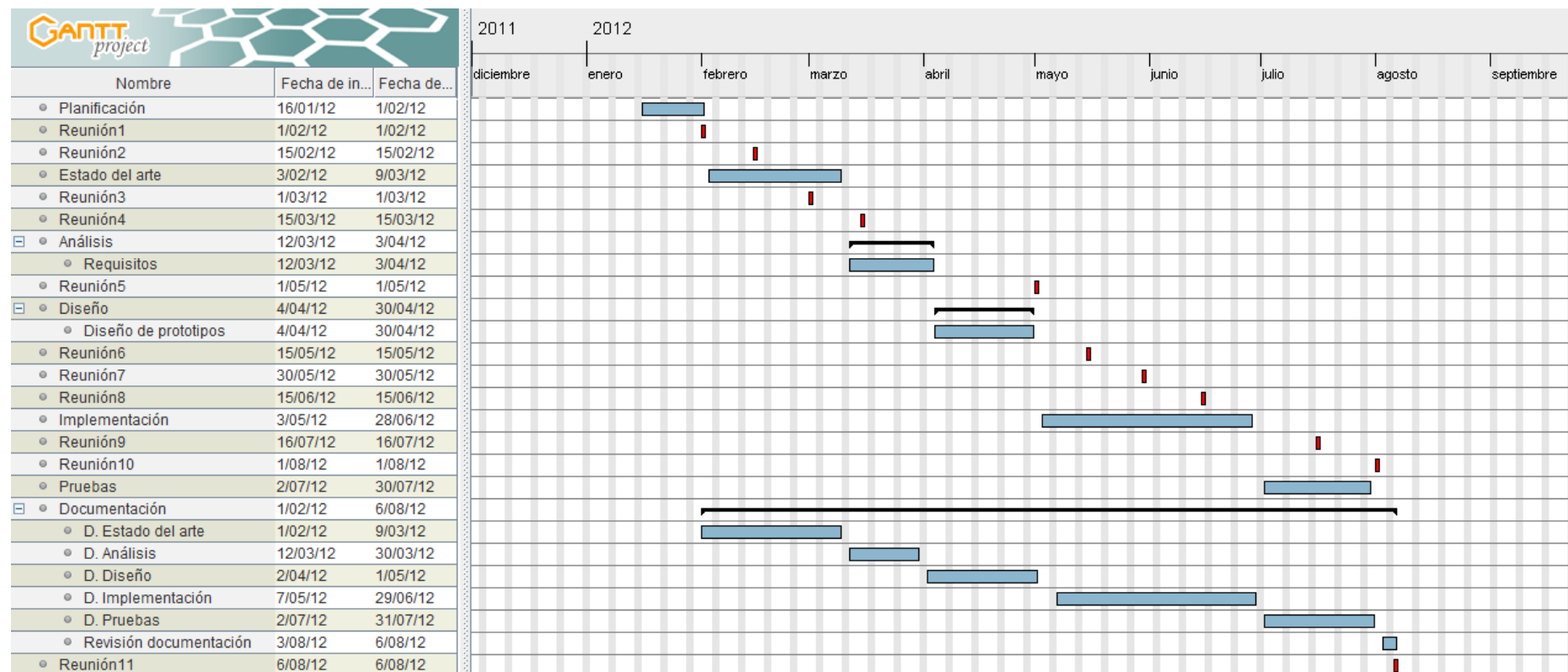


Ilustración 49 - Diagrama de Gantt inicial

Cabe destacar que los puntos coloreados en rojo representan las mencionadas reuniones de seguimiento. Como se puede observar se realizan en algunas ocasiones de forma quincenal y en otras de forma mensual, dependiendo de la carga de trabajo planificada.

## 6.1.2 Planificación final

La planificación inicial ha sufrido varios cambios a lo largo del desarrollo del proyecto. Esto ha supuesto un cambio significativo, replanificando el tiempo de trabajo dedicado a varias de las tareas planificadas. Dichos cambios y los factores que han intervenido para variar la planificación inicial han sido los siguientes:

La fase de planificación se ha completado según previsto, realizando las tareas en el tiempo de trabajo especificado. Por otro lado, la fase de análisis ha sufrido algún que otro cambio, pero cumpliendo con los plazos establecidos.

El cambio significativo que ha sufrido esta planificación ha empezado en Abril, ya que en un principio se pensó que se seguiría trabajando con el mismo ritmo, pero este fue interrumpido por la carga de trabajo de las asignaturas del último cuatrimestre de la carrera.

Todo esto ha supuesto un retraso en todas las tareas a partir del análisis. La pérdida de este mes en mitad del desarrollo del proyecto ha supuesto a parte de un retraso lógico de un mes de finalización del mismo ciertos problemas con la compatibilidad de dicho proyecto con el estudio de exámenes finales del cuatrimestre.

Aun con todos estos inconvenientes y el retraso se ha cumplido con los plazos esperados tanto en diseño como en implementación, únicamente que se ha desplazado un mes.

En cuanto al plan de pruebas se ha realizado en el menor tiempo posible para poder organizar todos antes de la entrega del día 5 de septiembre.

Finalmente, la fase de documentación ha sido la que menos cambios en cuanto a la planificación inicial ha sufrido. Esto es debido a que la redacción del presente documento diera comienzo el primer día de trabajo, tal y cómo se refleja en la planificación inicial. Y destacar que las reuniones han sido muy parecidas a las previstas.

A continuación se muestra las tareas y el diagrama de Gantt que refleja la planificación final del proyecto:

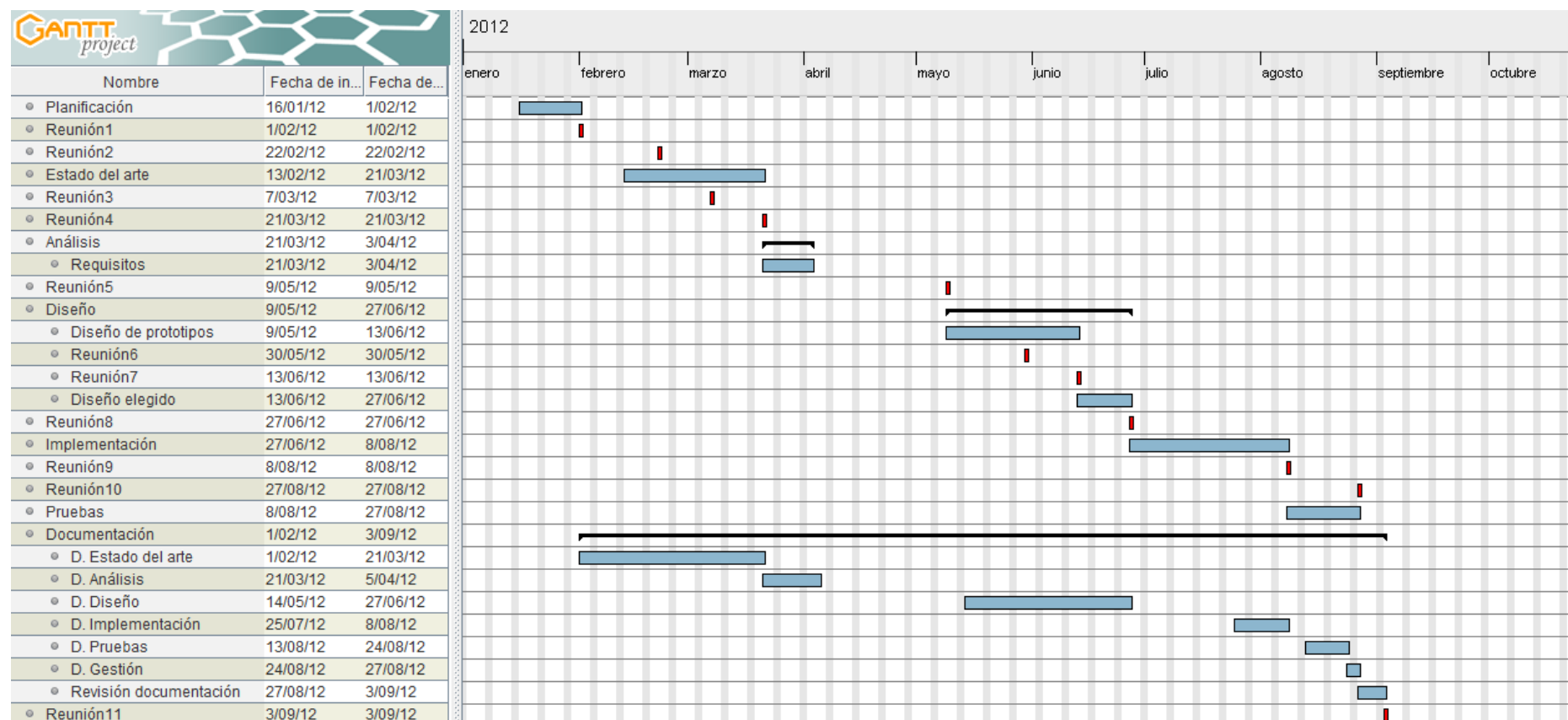


Ilustración 50 - Diagrama de Gantt final



## 6.2 Análisis económico

Una vez conocido el esfuerzo necesario para la realización del proyecto, se pasa a detallar el coste del mismo. En primer lugar se computan los gastos asociados al personal que se encarga de la realización de la aplicación. Según el cargo que ocupe su salario varía. Tras esto se explica detalladamente los gastos de hardware y de software necesarios para la realización del proyecto. Además se contabilizan también otros tipos de gastos que conlleva realizar todas las acciones correspondientes al proyecto, como por ejemplo la luz empleada en los sistemas para el funcionamiento. Para finalizar, se contabilizará todos los gastos y se dará una cifra exacta del precio de la aplicación.

### 6.2.1 Coste personal

Como personal se ha incluido a Álvaro García Reyes como principal desarrollador de la aplicación. También a Javier García Guzmán, ya que es el tutor del proyecto y se deben computar sus horas de reuniones y ayuda en el desarrollo de la aplicación.

A continuación se mostrará una tabla con todos los detalles.

Personal	Categoría	Coste hombre/hora	Dedicación (horas)	Coste (euros)
García Reyes, Álvaro	Ingeniero Junior	26	346	8 996
García Guzmán, Javier	Ingeniero Senior	38	16,5	627
Hombre hora		56	Total	<b>9 623</b>

Tabla 54 - Costes de personal

El coste mencionado ya incluye los valores del IRPF y Seguridad Social.

El coste total asociado al personal es de 9 623€

### 6.2.2 Coste Hardware

Ahora se calculará los gastos asociados al hardware utilizado. Para ello se debe tener en cuenta las amortizaciones de estos dispositivos. Para ello hay que prefijar la vida útil de cada dispositivo y la utilización de cada uno de ellos.

Se mostrará una tabla con todos los detalles referentes al gasto asociado. Aquellos detalles que no se reflejen en la tabla serán comentados a continuación de dicha tabla.

Concepto	Unidades	Precio Unitario (euros)	Vida útil Estimada (meses)	Tiempo de uso (meses)	Coste para el proyecto (euros)
Ordenador MacBook Pro 13"	1	1 281	48	2	53,38
Ordenador iMac 21.5"	1	1 174	48	5	122,3
iPhone 4S	1	399	36	2	20,47
Escáner Canon CanoScan LiDE 700F	1	129	24	5	26,88
Samsung Galaxy S i9000	1	149	36	1	4,14
<b>Total</b>					<b>227,17</b>

Tabla 55 - Coste de hardware

Ahora se explicará porqué de la estimación de la vida útil de cada dispositivo:

- MacBook Pro 13", se ha elegido dos meses debido a que este ordenador solo se ha utilizado en los meses de verano cuando no se disponía del iMac 21,5".
- iMac 21.5", se ha elegido cinco meses debido a que ha sido el ordenador que se ha utilizado durante la fase de desarrollo del proyecto, menos dos meses de verano.
- iPhone 4S, solamente se ha asignado dos meses debido a que han sido los suficientes como para realizar las pruebas pertinentes de la aplicación.
- Escáner Canon, se ha elegido cinco meses debido a que se ha utilizado a lo largo de todo el desarrollo menos en los dos primeros meses. Este escáner ha sido utilizado para la creación de las imágenes de la aplicación.
- Samsung Galaxy S, solamente se ha asignado un mes debido a que ha sido utilizado únicamente para realizar las búsquedas pertinentes en el Google play.

Todos los precios están redondeados a la alza en centésimas de euro. Y está incluido el IVA.

El coste total asociado al hardware es de 227,17 €

### 6.2.3 Coste Software

Ahora se calculará el gasto asociado al software utilizado a lo largo del proyecto. Además se adjuntará una tabla para poder observar dicho gasto. A continuación de dicha tabla se explicará que licencias ya vienen incluidas en los dispositivos hardware adquiridos y cuales hay que adquirir la licencia de uso.

Concepto	Unidades	Precio Unitario (euros)	Coste para el proyecto (euros)
MAC OS X Mountain Lion 10.6.8	1	0	0
IOS 5	1	0	0
Xcode 4.3.3	1	0	0
Adobe Photoshop CS6 Extended Student and Teacher Edition	1	199	199
Microsoft Office hogar y estudiantes 2010	1	99	99
StartUML versión: 5.0.2.1570	1	0	0
GanttProject 2.5.5 Praha	1	0	0
OmniGraffle Professional	1	0	0
<b>Total</b>			<b>298</b>

Tabla 56 - Coste de software

El software que ya viene incluido en el hardware adquirido es el siguiente:

- MAC OS X Mountain Lion 10.6.8 está integrado tanto en el iMac 23” como en el MacBook Pro 13”, por lo tanto su coste es de 0 €.
- IOS 5 está integrado en el iPhone 4S adquirido.
- Xcode 4.3.3 es gratuito desde el Apple Store.

Además hay algunos programas de código libre que son gratuitos y son los siguientes:

- StartUML versión: 5.0.2.1570
- GanttProject 2.5.5 Praha

Por último el OmniGraffle Professional se a utilizado en el departamento de informática de la universidad por lo que no ha supuesto ningún gasto extra, sino que este coste sería indirecto.

El IVA está incluido en el precio del producto.

El coste total asociado al software es de 298 €

## 6.2.4 Costes indirectos

Se ha decido añadir un cierto porcentaje de costes indirectos ya que existen ciertos costes que no se puede contabilizar de una manera exacta: como la luz del despacho asignado a este proyecto, el coste asociado a la conexión a internet, el programa OmniGraffle Professional para todo el personal del departamento, al igual que la licencia para subir la aplicación al Apple Store, etcétera.

Por lo tanto se ha decidido aplicar un 20% de los costes totales.

Tipo	Porcentaje (%)	Coste (euros)
Costes		10 148,17
Costes indirectos	20	<b>2 029,64</b>

Tabla 57 – Costes indirectos

El coste total asociado a los costes indirectos es de 2 029,64 €

## 6.2.5 Costes totales

Por último a continuación se muestra el coste final de la aplicación, sumando todos los costes obtenidos anteriormente.

Presupuesto	Coste (euros)
Coste personal	9 623
Coste hardware	227,17
Coste software	298
Coste indirecto	2 029,64
<b>Coste total</b>	<b>12 177,81</b>

Tabla 58 – Coste Total

Así pues el coste total del proyecto asciende a **12 177,81 euros** (doce mil ciento setenta y siete euros con ochenta y un céntimos).

# Capítulo 7 Conclusiones

7.1 Conclusiones del proyecto.....	94
7.1.1 Líneas futuras .....	95
7.2 Conclusiones personales.....	96

A continuación se disponen las conclusiones obtenidas una vez finalizado el proyecto. En primer lugar se establecerán las conclusiones relacionadas con el proyecto en sí, posteriormente las conclusiones a nivel personal y para finalizar las líneas futuras para la ampliación y mejora de la aplicación.

## 7.1 Conclusiones del proyecto

Se ha desarrollado como trabajo fin de grado una aplicación móvil de un juego interactivo por medio de láminas de errores. En el cual los usuarios, los niños, intentarán acertar las posibles incidencias que haya en ellas y así aprenderán las normas básicas de seguridad vial.

El desarrollo de este trabajo fin de grado ha supuesto dos problemas destacados. El primero de ellos es el diseño de interfaces sencillas y atractivas para un público tan exigente como son los niños. El segundo de ellos es la puesta en marcha del proyecto sobre un entorno real.

El primer problema ha supuesto marcar nuevos objetivos como informático. A lo largo de la carrera en escasas ocasiones se ve cómo se diseña una interfaz. En ningún caso nos enseñan cómo crear una interfaz atractiva para el usuario. El diseño de las interfaces ha requerido la comprensión y el manejo de las herramientas de diseño gráfico como el Adobe Photoshop. Además el diseño ha supuesto el estudio de manuales y ejemplos de interfaces<sup>17</sup>.

El segundo problema implica la preparación de la aplicación para su despliegue, es decir, para ser usada por los usuarios. Es siempre necesario controlar todos los posibles errores. Para controlar todos los posibles casos se ha hecho hincapié en pruebas de aceptación. Todo ello para intentar quebrar el correcto funcionamiento de la aplicación. Además cabe destacar que a lo largo del proyecto se han hecho distintas “demos” para que los usuarios puedan probarlas y así asegurar el perfecto funcionamiento de la aplicación.

La planificación inicial se ha visto postergada debido a factores que han aparecido a lo largo del desarrollo. A medida que se iba aprendiendo el lenguaje de programación<sup>9</sup> se iba cogiendo confianza y descubriendo nuevas funcionalidades<sup>10</sup> y métodos de hacer las cosas de una manera mejor<sup>11</sup>. Se ha aprendido a desarrollar en un entorno nuevo y desconocido<sup>12</sup>, a investigar mejores maneras de programar... en definitiva, se han adquirido las habilidades necesarias para ser capaz de desarrollar un proyecto de manera eficaz.

En este proceso no todo han sido éxitos. Ha habido momentos en los que han surgido problemas. La mayor parte de ellos se han solucionado de una forma u otra. Cabe destacar que el desarrollo del proyecto se ha realizado a la par con la finalización de la carrera suponiendo un gran esfuerzo y trabajo.

En lo que respecta a los objetivos que se habían establecido para este TFG, se evaluará hasta qué punto se ha llegado a la consecución de los mismos. Para ello, se volverán a enumerar cada uno de ellos, adjuntando las conclusiones que se pueden obtener tras el desarrollo:

- Debe ser un juego. Este objetivo es la esencia de la aplicación móvil desarrollada. Está se basa en un juego interactivo con el usuario por medio de láminas de errores, dónde el usuario puede aprender las normas de seguridad vial básicas que se plasman en estas láminas.
- La aplicación contenga diferentes niveles de dificultad. Este objetivo se he podido llevar a cabo gracias a la creación de tres láminas diferentes. En las cuales a simple vista se puede ver que se va incrementando el nivel de dificultad de cada una.
- A ser posible que sea original. Este objetivo se cree que se ha cumplido gracias al empeño y dedicación puestos en el diseño de las imágenes de la aplicación. Se cree que es de lo más original y más orientado al público que va destinado esta aplicación.

Para finalizar comentar que se ha partido desde cero y se ha pasado por todas las fases de desarrollo para concluir la creación del proyecto. Muchos de los conocimientos adquiridos a lo largo de la carrera se han puesto de manifiesto al afrontar su desarrollo. Por tanto cabe decir que es una gran satisfacción ver cómo una idea puede llevarse a cabo si se trabaja con dedicación y con empeño.

### 7.1.1 Líneas futuras

Una vez expuestas las conclusiones, se pueden obtener una serie de líneas futuras que no se extraen de los objetivos pero que sería interesante desarrollar. Estas pueden mejorar las prestaciones de la aplicación para una mejor aceptación en el mercado.

## A. Game Center

Game Center es una red social de juegos online multijugador lanzado por Apple. Se permite a los usuarios invitar a amigos a jugar a un juego, iniciar un juego multijugador a través de contactos, seguimiento de sus logros, y comparar sus puntuaciones en una tabla de líderes<sup>15</sup>.

Game Center realmente es una revolución en los juegos de Apple, incluso la mayoría de los juegos que implementan Game Center ganan cierta reputación en cuanto a la opinión de los usuarios gracias a las prestación que permite.

Cabe destacar que se ha intentado a lo largo del proyecto realizar esta implementación, pero por falta de tiempo, lo extensible que es dicha implementación y las numerosas pruebas unitarias que habría que haber realizado, finalmente se ha descartado que esté en este proyecto. Aunque gran parte de la implementación está

adaptada al código realizado faltando nada más que la creación de logros y las numerosas pruebas unitarias.

## **B. Extensión al iPad**

Esta Tablet de Apple tiene cada vez más usuarios, por tanto un objetivo sería la adaptación del código de iPhone al iPad. Una tarea costosa pero no muy difícil en cuanto a la implementación. Algunos de los cambios de se deberían apreciar son los siguientes:

- El iPad ofrece una pantalla mucho mayor por lo que la interfaz gráfica se vería considerablemente modificada. Para implementar la extensión al iPad sería necesario rediseñar algunas interfaces ya que en la pantalla del iPad, las interfaces existentes quedarían demasiado holgadas.
- También habría que cambiar el tipo de vista de algunas clases para adaptarla al tipo de vistas en el iPad. Además se utilizaría una vista horizontal de las plantillas para que se visualice mejor cada detalle de las láminas.

## **C. Integración de una base de datos remota**

Como se ha explicado a lo largo del proyecto integrar una base de datos remota podría aportar más cantidad de láminas e incluso más cantidad de modalidades de juego. Además de un posible beneficio con la adquisición por parte del usuario de los nuevos tipos de juego o de las nuevas láminas ya que la descarga del juego sería gratuita, por su afán didáctico y difusión al público más pequeño.

La integración de esta base de datos remota no sería una tarea fácil de realizar ya que la realización por parte del desarrollador es un trabajo extra al implementar todas las nuevas funcionalidades, el diseño y mantenimiento de la base de datos y la creación e innovación de nuevas láminas.

## **7.2 Conclusiones personales**

Cuando leí el título del proyecto por primera vez “Aplicación móvil para fomentar la seguridad vial en niños (plataforma IOS)”, fue inmediata la reacción positiva que tuve. Sabía que había encontrado el proyecto que quería desarrollar. Esto es debido a que a lo largo de estos cuatro años de la carrera me he dedicado a dar clases particulares a diferentes niños, desde los más pequeños (6 años) hasta alumnos que únicamente nos diferenciaban un par de años. Como no, se les coge un cariño especial a los más pequeños.

La primera impresión que tuve sobre este proyecto no ha cambiado en ningún momento a lo largo del desarrollo del mismo, todo lo contrario, cada vez que aparecían



nuevos retos e ideas se incrementaba mi interés por llevarlos a cabo, pensando en cada rato libre como poder mejorar lo pensado y qué cosas nuevas añadir.

Me sumergí en un mundo totalmente desconocido por mí como era la plataforma IOS. Jamás había desempeñado ninguna función en un ordenador Mac y mucho menos la programación de una aplicación. Esto supuso un gran reto de superación. Además intenté integrarme lo más posible en la creación de una aplicación única. Para ello tomé la decisión de crear por mí mismo todas las imágenes e interfaces de la aplicación. Suponiendo un gran reto personal y creativo, llevar a cabo todas esas imágenes, pero nada se puso en mi camino ya que realmente estaba ilusionado por llevar a cabo este proyecto.

La finalización de la aplicación y la memoria ha supuesto alcanzar un alto grado de satisfacción personal, ya que he conseguido sacar adelante un proyecto interesante y entretenido pasando por todas las fases de desarrollo software estudiadas en la carrera. Dicha satisfacción se debe de la misma forma al esfuerzo y horas empleadas para resolver problemas surgidos en la aplicación. Estos problemas y otros factores, como la carga de trabajo en las últimas asignaturas de la carrera, han ido transformando de alguna forma el proyecto en un reto personal, que finalmente ha finalizado de forma satisfactoria por el esfuerzo empleado y la ayuda recibida por parte de mis compañeros.

El resultado final tras el desarrollo de la aplicación a nivel personal ha sido ampliamente gratificante, proporcionando una serie de conocimientos que no se habían adquirido durante la carrera y que pueden ser de utilidad para la vida laboral en un futuro cercano.

# Bibliografía

- [1] Apple Online Store: <http://store.apple.com/es>
- [2] Google Play: <https://play.google.com/store>
- [3] Google: [www.google.es](http://www.google.es)
- [4] Educapeques:  
<http://www.educapeques.com/secc/educacion-vial-para-ninos-a-traves-de-juegos.html>
- [5] Educación Skoda:  
<http://ww2.autoscout24.es/especial/seguridad-vial-para-ninos/educacion-online/44439/61629/>
- [6] Junta de Castilla y León:  
<http://www.educa.jcyl.es/educacyl/cm/zonaalumnos/tkPopUp?pgseed=1140443665302&idContent=31497&locale>
- [7] Dirección general de tráfico:  
[http://aplch.dgt.es/pevi/faces/paginas/recursos/curriculares/did\\_primaria/detalle.xhtml?idDoc=060f4244800fd994&paramMenuSeleccionado=submenu\\_1\\_1](http://aplch.dgt.es/pevi/faces/paginas/recursos/curriculares/did_primaria/detalle.xhtml?idDoc=060f4244800fd994&paramMenuSeleccionado=submenu_1_1)
- [8] John Ray: “Sams Teach Yourself iPhone Application Development in 24 Hours”, Second Edition. Sams. 2010. [En línea]  
<http://proquest.safaribooksonline.com/book/programming/mobile/9780132601214>
- [9] Erica Sadun: “The iOS 5 Developer’s Cookbook: Core Concepts and Essential Recipes for iOS Programmers”, Third Edition. Addison-Wesley Professional. 2011. [En línea]  
<http://proquest.safaribooksonline.com/book/programming/mobile/9780132613477>
- [10] Matt Neuberg: “Programming iOS 5”, Second Edition. O’Reilly Media, Inc. 2012. [En línea]  
<http://proquest.safaribooksonline.com/book/programming/mobile/9781449327088>
- [11] Joe Conway and Aaron Hillegass: “iOS Programming: The Big Nerd Ranch Guide”, Big Nerd Ranch Guides. 2012. [En línea]  
<http://proquest.safaribooksonline.com/book/-/9780132978767>
- [12] Frederic F. Anderson: “Xcode® 4 Unleashed”, Second Edition. Sams. 2012. [En línea]  
<http://proquest.safaribooksonline.com/book/software-engineering-and-development/xcode/9780768697537>

- [13] Steffen Itterheim and Andreas Löw: “Learn cocos2D Game Development with iOS 5” Apress. 2011. [En línea]  
<http://proquest.safaribooksonline.com/book/programming/game-programming/9781430238133>
- [14] James Sugrue: “Building iOS 5 Games: Develop and Design” Peachpit Press. 2011. [En línea]  
<http://proquest.safaribooksonline.com/book/programming/game-programming/9780132790451>
- [15] Vanda Nahavandipour: “Writing Game Center Apps in iOS” O'Reilly Media, Inc. 2011. [En línea]  
<http://proquest.safaribooksonline.com/book/programming/game-programming/9781449307950>
- [16] SOFTENG. Metodología Scrum [En línea]  
<http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>
- [17] Steve Caplin: “100% Photoshop” Focal Press. 2010. [En línea]  
<http://proquest.safaribooksonline.com/book/-/9780240814254>